

PCT/JP00/07549

JP00/7549

EU

日 本 国 特 許 庁

PATENT OFFICE
JAPANESE GOVERNMENT

01 12.00

REC'D 26 JAN 2001

WIPO PCT

別紙添付の書類に記載されている事項は下記の出願書類に記載されて
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office.

出 願 年 月 日

Date of Application:

2000年10月25日

出 願 番 号

Application Number:

特願2000-325218

出 願 人

Applicant (s):

ソニー株式会社

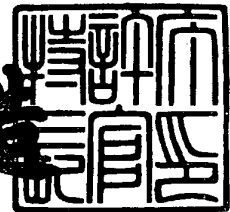
PRIORITY
DOCUMENT

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)

2001年 1月12日

特許庁長官
Commissioner,
Patent Office

及 川 耕 造



出証番号 出証特2000-3110988

【書類名】 特許願

【整理番号】 0000879904

【提出日】 平成12年10月25日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 13/00

【発明者】

【住所又は居所】 東京都品川区北品川6丁目7番35号 ソニー株式会社
内

【氏名】 岩津 健

【特許出願人】

【識別番号】 000002185

【氏名又は名称】 ソニー株式会社

【代表者】 出井 伸之

【代理人】

【識別番号】 100082131

【弁理士】

【氏名又は名称】 稲本 義雄

【電話番号】 03-3369-6479

【先の出願に基づく優先権主張】

【出願番号】 平成11年特許願第308673号

【出願日】 平成11年10月29日

【手数料の表示】

【予納台帳番号】 032089

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9708842

特 2 0 0 0 - 3 2 5 2 1 8

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 記憶媒体、情報処理装置および方法

【特許請求の範囲】

【請求項1】 制御部とハードウェアとの間に位置し、上記制御部からのメッセージに基づいて上記ハードウェアを制御するプログラムは、

上記制御部と通信する第1のプロセスと、

第1のハードウェアおよび第2のハードウェアと通信可能な第2のプロセスと、

上記第1のプロセスおよび上記第2のプロセスと通信すると共に、上記第1のハードウェアに対応するインタフェース処理を実行する第3のプロセスと、

上記第1のプロセスおよび上記第2のプロセスと通信すると共に、上記第2のハードウェアに対応するインターフェース処理を実行する第4のプロセスとを備え、

上記第1のプロセスは、上記制御部からのメッセージに基づいて、上記第3のプロセスおよび前記第4のプロセスのいずれか一方にメッセージを出力するコンピュータ制御可能なプログラムが記憶されている記憶媒体。

【請求項2】 請求項1において、

上記プログラムは、更に、

上記制御部と通信する第5のプロセスを備え、

上記第5のプロセスは、上記制御部からのメッセージに応じて、上記ハードウェアが処理すべきデータを取得するコンピュータ制御可能なプログラムが記憶されている記憶媒体。

【請求項3】 請求項2において、

上記制御部は、メインアプリケーションプロセスを含むコンピュータ制御可能なプログラムが記憶されている記憶媒体。

【請求項4】 請求項3において、

上記プログラムは、

上記メインアプリケーションプロセスからのデータ格納場所を指定するメッセージを第5のプロセスが受信し、

上記第5のプロセスは、上記データ格納場所に応じて上記ハードウェアが処理すべきデータを取得し、

上記メインアプリケーションプロセスからのエンコード方式を指定するメッセージを第1のプロセスが受信し、

上記エンコード方式に応じて、上記第3のプロセスおよび上記第4のプロセスのいずれか一方が、上記第1のプロセスから上記メッセージを受信し、

上記第3のプロセスおよび上記第4のプロセスのいずれか一方の、上記第2のプロセスとの通信に応じて、上記第2のプロセスと通信する第1 或いは第2のハードウェアが上記取得されたデータをエンコードするコンピュータ制御可能なプログラムが記憶されている記憶媒体。

【請求項5】 請求項4において、

上記プログラムは、

上記メインアプリケーションプロセスからのデータ格納場所を指定するメッセージをデータ入出力マネージャプロセスが受信し、

上記データ入出力マネージャプロセスは、上記データ格納場所に応じて上記ハードウェアが処理すべきデータを取得し、

上記メインアプリケーションプロセスからのエンコード方式を指定するメッセージをエンコード処理マネージャプロセスが受信し、

上記エンコード方式に応じて、第1のエンコードカード入出力I/Fプロセスおよび第2のエンコードカード入出力I/Fプロセスのいずれか一方が、上記エンコード処理マネージャプロセスから上記メッセージを受信し、

上記第1のエンコードカード入出力I/Fプロセスおよび第2のエンコードカード入出力I/Fプロセスのいずれか一方の、エンコードカードドライバプロセスとの通信に応じて、上記エンコードカードドライバプロセスと通信する第1 或いは第2のハードウェアが上記取得されたデータをエンコードするコンピュータ制御可能なプログラムが記憶されている記憶媒体。

【請求項6】 請求項5において、

上記エンコードカードドライバプロセスは、ATRAC方式でオーディオデータをエンコードする第1のエンコーダ、および

MPEG Audio Layer 3 方式でオーディオデータをエンコードする第2のエンコーダと通信可能であるコンピュータ制御可能なプログラムが記憶されている記憶媒体。

【請求項7】 請求項1において、

上記各プロセスには、優先権が設定されているコンピュータ制御可能なプログラムが記憶されている記憶媒体。

【請求項8】 請求項1において、

上記各プロセスは、異常時には、

生成されるとき遷移する第1のステートを含む第1のパス、

終了するとき遷移する第2のステート、生成されるとき遷移する第3のステート、および上記第1のステートを含む第2のパス、または

上記第2のステート、および終了するとき遷移する第4のステートを含む第3のパス

のいずれかのパスによりステートを遷移するコンピュータ制御可能なプログラムが記憶されている記憶媒体。

【請求項9】 制御部とソフトウェアとの間に位置し、上記制御部からのメッセージに基づいて上記ソフトウェアを制御するプログラムは、

上記制御部と通信する第1のプロセスと、

第1のソフトウェアエンコーダおよび第2のソフトウェアエンコーダと通信可能な第2のプロセスと、

上記第1のプロセスおよび上記第2のプロセスと通信すると共に、上記第1のソフトウェアエンコーダに対応するインターフェース処理を実行する第3のプロセスと、

上記第1のプロセスおよび上記第2のプロセスと通信すると共に、上記第2のソフトウェアエンコーダに対応するインターフェース処理を実行する第4のプロセスと

を備え、

上記第1のプロセスは、上記制御部からのメッセージに基づいて、上記第3のプロセスおよび前記第4のプロセスのいずれか一方にメッセージを出力するコン

コンピュータ制御可能なプログラムが記録されている記憶媒体。

【請求項 1 0】 制御部とハードウェアとの間に位置し、上記制御部からのメッセージに基づいて上記ハードウェアを制御する情報処理装置において、

上記制御部と通信する第 1 の手段と、

第 1 のハードウェアおよび第 2 のハードウェアと通信可能な第 2 の手段と、

上記第 1 の手段および上記第 2 の手段と通信すると共に、上記第 1 のハードウェアに対応するインタフェース処理を実行する第 3 の手段と、

上記第 1 の手段および上記第 2 の手段と通信すると共に、上記第 2 のハードウェアに対応するインターフェース処理を実行する第 4 の手段とを備え、

上記第 1 の手段は、上記制御部からのメッセージに基づいて、上記第 3 の手段および上記第 4 の手段のいずれか一方にメッセージを出力する情報処理装置。

【請求項 1 1】 請求項 1 0 において、

更に、

上記制御部と通信する第 5 の手段を備え、

上記第 5 の手段は、上記制御部からのメッセージに応じて、上記ハードウェアが処理すべきデータを取得する情報処理装置。

【請求項 1 2】 請求項 1 1 において、

上記第 5 の手段は、上記制御部からデータ格納場所を指定するメッセージを受信すると共に、上記データ格納場所に応じて上記ハードウェアが処理すべきデータを取得し、

上記第 1 の手段は、上記制御部からのエンコード方式を指定するメッセージを受信し、

上記エンコード方式に応じて、上記第 3 の手段および上記第 4 の手段のいずれか一方が、上記第 1 の手段から上記メッセージを受信し、

上記第 3 の手段および上記第 4 の手段のいずれか一方の、上記第 2 の手段との通信に応じて、上記第 2 の手段と通信する第 1 或いは第 2 のハードウェアが上記取得されたデータをエンコードする情報処理装置。

【請求項 1 3】 制御部とハードウェアとの間に位置し、上記制御部からの

メッセージに基づいて上記ハードウェアを制御する情報処理方法において、

上記制御部からのデータ格納場所を指定するメッセージに応じて、上記ハードウェアが処理すべきデータを取得するステップと、

上記制御部からのエンコード方式を指定するメッセージに応じて、第1のエンコーダ、或いは第2のエンコーダに対応するインタフェース処理を実行するステップと、

上記インタフェース処理されたメッセージを上記第1のエンコーダ、或いは上記第2のエンコーダに送信するステップと、

上記インタフェース処理されたメッセージに応じて、上記第1のエンコーダ、或いは上記第2のエンコーダにより上記取得されたデータをエンコードするステップとを備える

情報処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、記憶媒体、情報処理装置および方法に関し、特に、情報を処理するために必要なハードウェアの変更または改良が容易にできるようにした記憶媒体、情報処理装置および方法に関する。

【0002】

また、より迅速に異常なステートを回復することができるようにした記憶媒体、情報処理装置および方法に関する。

【0003】

【従来の技術】

図1には、従来の音楽配信サービスシステムの配信側の装置を構成するエンコーダ10の機能的構成例（エンコード処理用のプログラムの構成例）が示されている。なお、この音楽配信サービスシステムの配信側の装置は、このエンコーダ10の他、エンコーダ10におけるエンコード処理を制御する制御端末（図示せず）、およびエンコードされる音楽データ（以下 PCM (Pulse Code Modulation) 非圧縮音楽データと称する）をエンコーダ10に供給したり、エンコーダ10

によりエンコードされた音楽データ（以下、PCM圧縮音楽データと称する）を記憶し、音楽配信サービスの利用者（受信側）に配信するサーバ（図示せず）から構成されている。

【0004】

エンコーダ10には、エンコード処理を制御するための制御データが転送されるアラームLAN (alarm Local Area Network) との通信を制御するネットワークカード、PCM非圧縮音楽データおよびPCM圧縮音楽データ（以下、これらを個々に区別する必要がない場合、単に、音楽データと称する）が転送されるメディアムLAN (medium LAN) との通信を制御するネットワークカード、およびエンコード処理を実行するエンコードカードなどのハードウェアが設けられている。

【0005】

図1に示すエンコード処理用のプログラムは、制御データ入出力プロセス11、ネットワークカードドライバプロセス12、メイン処理プロセス13、ネットワークカードドライバプロセス14、およびエンコードカードドライバプロセス15の5つのプロセス（図中、実線の枠で示されている要素）から構成されている。なお、プロセスは、実行可能な所定のプログラムで構成され、データ領域（バッファやレジスタなど）を管理し、各処理は、このプロセス単位で実行される。

【0006】

制御データ入出力プロセス11は、メイン処理プロセス13と通信を行う。制御データ入出力プロセス11は、ネットワークカードドライバプロセス12を介して、制御端末から送信されてくる、エンコード処理に関する各種コマンドを受信し、メイン処理プロセス13に出力する。制御データ入出力プロセス11はまた、メイン処理プロセス13から供給される、例えば、エンコード処理が成功したことを示すメッセージやエンコード処理が失敗したことを示すエラーメッセージなどを、ネットワークカードドライバプロセス12を介して制御端末に出力する。なお、以下において、制御データ入出力プロセス11およびメイン処理プロセス13との間で授受されるコマンドやエラーメッセージを、個々に区別する必要がない場合、単に、制御データと称する。

【0007】

メイン処理プロセス13は、ネットワークカードドライバプロセス14を介して、サーバ（図示せず）から送信されてくる、例えば、PCM非圧縮音楽データを受信し、エンコードカードドライバプロセス15を介して、エンコードカードに供給する。メイン処理プロセス13はまた、エンコードカードドライバプロセス15を介してエンコードカードを制御し、制御データ入出力プロセス11からの制御データに基づくエンコード処理をエンコードカードに実行させる。この例の場合、エンコードカードにおいて、サンプリング周波数が48KHzであるMPEG (Moving Picture Experts Group) 1 layer 2に準拠したエンコード処理（以下、MPEG 1対応エンコード処理と称する）、またはサンプリング周波数が44.1KHzであるATRAC(adaptive transform acoustic coding) 1（商標）の規格に準拠したエンコード処理（以下、ATRAC 1対応エンコード処理と称する）が実行される。

【0008】

メイン処理プロセス13は、エンコード処理を施して得られたPCM圧縮音楽データを、エンコードカードドライバプロセス15を介して、エンコードカードから受け取り、それを、ネットワークカードドライバプロセス14を介してサーバに供給する。

【0009】

次に、制御データ入出力プロセス11およびメイン処理プロセス13について説明する。制御データ入出力プロセス11は、制御部21、ネットワークカード入力I/F(インターフェース)22、およびネットワークカード出力I/F23の3つのプログラム（図中、点線の枠で示されている要素）、並びにそれらを実行するために必要なデータ領域を有している。

【0010】

ネットワークカード入力I/F22は、ネットワークカードドライバプロセス12により受信された制御端末からの制御データを受け取り、制御部21に出力する。ネットワークカード出力I/F23は、制御部21を介して供給される、メイン処理プロセス13からの、例えば、エラーメッセージを受け取り、ネットワークカードドライバプロセス12に出力する。

【0011】

制御部21は、ネットワークカード入力I/F22およびネットワークカード出力I/F23を制御するとともに、メイン処理プロセス13の制御部31と通信を行う。

【0012】

次に、メイン処理プロセス13の構成について説明する。メイン処理プロセス13は、制御部31、ネットワークカード入出力I/F32、エンコードエンジン入出力I/F33、エンコードエンジン入出力I/F34、エンコードエンジン35、エンコードカード入出力I/F36、およびエンコードカード入出力I/F37の7つのプログラム（図中、点線の枠で示されている要素）、並びにそれらを実行させるために必要なデータ領域を有している。

【0013】

ネットワークカード出力I/F32は、ネットワークカードドライバプロセス14により受信されたPCM非圧縮音楽データを受け取り、制御部31に出力したり、制御部31を介して供給される、PCM圧縮音楽データを、ネットワークカードドライバプロセス14に出力する。

【0014】

エンコードエンジン入出力I/F33は、制御部31を介して入力されたPCM非圧縮音楽データのうち、ATRAC1対応エンコード処理されるPCM非圧縮音楽データを受け取り、エンコードエンジン35に出力する。エンコードエンジン入出力I/F33はまた、エンコードエンジン35から入力されるATRAC1対応エンコード処理されたPCM圧縮音楽データを、制御部31に出力する。

【0015】

エンコードエンジン入出力I/F34は、制御部31を介して入力されたPCM非圧縮音楽データのうち、MPEG1対応エンコード処理されるPCM非圧縮音楽データを受け取り、エンコードエンジン35に出力する。エンコードエンジン入出力I/F34はまた、エンコードエンジン35から入力されるMPEG1対応エンコード処理が施されたPCM圧縮音楽データを、制御部31に出力する。

【0016】

エンコードカード入出力I/F36は、制御部31を介して入力されたPCM非圧縮音楽データのうち、ATRAC1対応エンコード処理されるPCM非圧縮音楽データを受け取り、エンコードカードドライバプロセス15に出力する。エンコードカード入出力I/F36はまた、エンコードカードドライバプロセス15から入力されるATRAC1対応エンコード処理されたPCM圧縮音楽データを、制御部31に出力する。

【0017】

エンコードカード入出力I/F37は、制御部31を介して入力されたPCM非圧縮音楽データのうち、MPEG1対応エンコード処理されるPCM非圧縮音楽データを受け取り、エンコードカードドライバプロセス15に出力する。エンコードカード入出力I/F37はまた、エンコードカードドライバプロセス15から入力されるMPEG1対応エンコード処理が施されたPCM圧縮音楽データを、制御部31に出力する。

【0018】

エンコードエンジン35は、エンコードエンジン入出力I/F33, 34から供給されたPCM非圧縮音楽データを、ソフトウェア的に、ATRAC1対応エンコード処理、またはMPEG1対応エンコード処理し、得られたPCM圧縮音楽データを、それぞれに戻す。

【0019】

エンコーダ10は以上のように構成されているが、制御データ入出力プロセス11乃至エンコードカードドライバプロセス15を構成する各プログラムは、エンコーダ10に設けられた、ネットワークカードやエンコードカードなどの所定のハードウェアに依存して構築されている。

【0020】

従って、例えばインターネット、デジタル衛星放送、地上波デジタル放送等の伝送媒体に対応して、エンコーダ10に設けられているネットワークカードやエンコードカードを変更したり、拡張したりする場合、各プロセスを再構築する必要があり、時間および費用がかかる課題があった。

【0021】

また、例えば、エンコーダ 10 に設けられたハードウェアとしてのネットワークカードやエンコードカードを制御するインターフェースプログラム（ネットワークカード入力 I/F 22、ネットワークカード出力 I/F 23、ネットワークカード入出力 I/F 32 乃至エンコードカード入出力 I/F 37）およびドライバプログラム（ネットワークカードドライバプロセス 12、14、およびエンコードカードドライバプロセス 15 を構成するプログラム）は、通常、エンコーダ 10 の製造業者により作成される。

【0022】

【発明が解決しようとする課題】

しかしながら、エンコーダ 10 の 1 つの部品としてのハードウェアの機能がより高性能になると、部品としてのハードウェアを組み込んでエンコーダ 10 を製造する製造業者が、部品としてのハードウェアのインターフェースプログラムやドライバプログラムを直接作成することは、非効率的となり、ハードウェアの製造業者（部品の製造業者）が各プログラムを作成し、ハードウェアとともにエンコーダ 10 の製造業者に提供することが多くなる。

【0023】

このような場合、エンコーダ 10 の製造業者は、部品としてのハードウェアの詳細は判らなくなるため、それを制御する制御部 21 や制御部 31 の作成が困難になり、ますます、ハードウェアの変更や拡張も困難になる課題があった。

【0024】

更に、上述のようにエンコーダ 10 は構成されているが、プログラムの処理に異常が発生した場合、プログラムの処理を一旦終了させるようにしているので、処理を再開させるのに時間がかかる課題があった。

【0025】

本発明はこのような状況に鑑みてなされたものであり、ハードウェアの変更や拡張が容易にできるようにするものである。

【0026】

更に、プログラムの処理の異常を迅速に回復することができるようにするものである。

【 0 0 2 7 】

【課題を解決するための手段】

本発明のコンピュータ制御可能なプログラムが記憶されている記憶媒体は、制御部とハードウェアとの間に位置し、制御部からのメッセージに基づいてハードウェアを制御するプログラムが、制御部と通信する第1のプロセスと、第1のハードウェアおよび第2のハードウェアと通信可能な第2のプロセスと、第1のプロセスおよび第2のプロセスと通信すると共に、第1のハードウェアに対応するインタフェース処理を実行する第3のプロセスと、第1のプロセスおよび第2のプロセスと通信すると共に、第2のハードウェアに対応するインターフェース処理を実行する第4のプロセスとを備え、第1のプロセスは、制御部からのメッセージに基づいて、第3のプロセスおよび前記第4のプロセスのいずれか一方にメッセージを出力する。

【 0 0 2 8 】

上記プログラムは、更に、制御部と通信する第5のプロセスを備え、第5のプロセスは、制御部からのメッセージに応じて、ハードウェアが処理すべきデータを取得するようにすることができる。

【 0 0 2 9 】

上記制御部は、メインアプリケーションプロセスを含むようにすることができる。

【 0 0 3 0 】

上記プログラムは、メインアプリケーションプロセスからのデータ格納場所を指定するメッセージを第5のプロセスが受信し、第5のプロセスは、データ格納場所に応じてハードウェアが処理すべきデータを取得し、メインアプリケーションプロセスからのエンコード方式を指定するメッセージを第1のプロセスが受信し、エンコード方式に応じて、第3のプロセスおよび第4のプロセスのいずれか一方が、第1のプロセスからメッセージを受信し、第3のプロセスおよび第4のプロセスのいずれか一方の、第2のプロセスとの通信に応じて、第2のプロセスと通信する第1或いは第2のハードウェアが取得されたデータをエンコードするようにすることができる。

【0031】

上記プログラムは、メインアプリケーションプロセスからのデータ格納場所を指定するメッセージをデータ入出力マネージャプロセスが受信し、データ入出力マネージャプロセスは、データ格納場所に応じてハードウェアが処理すべきデータを取得し、メインアプリケーションプロセスからのエンコード方式を指定するメッセージをエンコード処理マネージャプロセスが受信し、エンコード方式に応じて、第1のエンコードカード入出力I/Fプロセスおよび第2のエンコードカード入出力I/Fプロセスのいずれか一方が、エンコード処理マネージャプロセスからメッセージを受信し、第1のエンコードカード入出力I/Fプロセスおよび第2のエンコードカード入出力I/Fプロセスのいずれか一方の、エンコードカードドライバプロセスとの通信に応じて、エンコードカードドライバプロセスと通信する第1或いは第2のハードウェアが取得されたデータをエンコードするようにすることができるようにすることができる。

【0032】

上記エンコードカードドライバプロセスは、ATRAC方式でオーディオデータをエンコードする第1のエンコーダ、およびMPEG Audio Layer 3方式でオーディオデータをエンコードする第2のエンコーダと通信可能であるようにすることができる。

【0033】

上記各プロセスには、優先権が設定されているようにすることができる。

【0034】

上記各プロセスは、異常時には、生成されるとき遷移する第1のステートを含む第1のパス、終了するとき遷移する第2のステート、生成されるとき遷移する第3のステート、および第1のステートを含む第2のパス、または第2のステート、および終了するとき遷移する第4のステートを含む第3のパスのいずれかのパスによりステートを遷移するようにすることができる。

【0035】

本発明のコンピュータ制御可能なプログラムが記録されている記憶媒体は、制御部とソフトウェアとの間に位置し、制御部からのメッセージに基づいてソフト

ウェアを制御するプログラムが、制御部と通信する第1のプロセスと、第1のソフトウェアエンコーダおよび第2のソフトウェアエンコーダと通信可能な第2のプロセスと、第1のプロセスおよび第2のプロセスと通信すると共に、第1のソフトウェアエンコーダに対応するインタフェース処理を実行する第3のプロセスと、第1のプロセスおよび第2のプロセスと通信すると共に、第2のソフトウェアエンコーダに対応するインターフェース処理を実行する第4のプロセスとを備え、第1のプロセスは、制御部からのメッセージに基づいて、第3のプロセスおよび前記第4のプロセスのいずれか一方にメッセージを出力する。

【0036】

本発明の情報処理装置は、制御部と通信する第1の手段と、第1のハードウェアおよび第2のハードウェアと通信可能な第2の手段と、第1の手段および第2の手段と通信すると共に、第1のハードウェアに対応するインタフェース処理を実行する第3の手段と、第1の手段および第2の手段と通信すると共に、第2のハードウェアに対応するインターフェース処理を実行する第4の手段とを備え、第1の手段は、制御部からのメッセージに基づいて、第3の手段および第4の手段のいずれか一方にメッセージを出力する。

【0037】

更に、制御部と通信する第5の手段を備え、第5の手段は、制御部からのメッセージに応じて、ハードウェアが処理すべきデータを取得するようにすることができる。

【0038】

上記第5の手段は、制御部からデータ格納場所を指定するメッセージを受信すると共に、データ格納場所に応じてハードウェアが処理すべきデータを取得し、第1の手段は、制御部からのエンコード方式を指定するメッセージを受信し、エンコード方式に応じて、第3の手段および第4の手段のいずれか一方が、第1の手段からメッセージを受信し、第3の手段および第4の手段のいずれか一方の、第2の手段との通信に応じて、第2の手段と通信する第1或いは第2のハードウェアが取得されたデータをエンコードするようにすることができる。

【0039】

本発明の情報処理方法は、制御部からのデータ格納場所を指定するメッセージに応じて、ハードウェアが処理すべきデータを取得するステップと、制御部からのエンコード方式を指定するメッセージに応じて、第1のエンコーダ、或いは第2のエンコーダに対応するインタフェース処理を実行するステップと、インタフェース処理されたメッセージを第1のエンコーダ、或いは第2のエンコーダに送信するステップと、インタフェース処理されたメッセージに応じて、第1のエンコーダ、或いは第2のエンコーダにより取得されたデータをエンコードするステップとを備える。

【0040】

従って、いずれの場合においても、ハードウェアが変更されたとしても、制御部を変更する必要がなくなるか、変更するとしても、わずかの変更で済み、1つの制御部で、多くのハードウェアを共通に使用することが可能になる。

【0041】

【発明の実施の形態】

図2は、本発明を適用した音楽配信サービスシステムの配信側の構成例を示している。この例の場合、例えば、100BASE-TXイーサネット（商標）からなる、アラームLAN101およびメディアムLAN102が設けられている。アラームLAN101には、端末111およびエンコーダ112が接続され、メディアムLAN102には、エンコーダ112およびサーバ113が接続されている。

【0042】

端末111は、エンコーダ112を制御するためのプログラムからなる制御プロセスWを有しており、その制御プロセスWに基づいて、アラームLAN101を介してエンコーダ112と通信して、エンコーダ112を制御し、所定のエンコード処理を実行させる。

【0043】

エンコーダ112は、端末111により制御され、サーバ113に保持されている、CD-DA(Compact Disc-Digital Audio)などのPCM非圧縮音楽データを、メディアムLAN102を介して受信し、ATRAC1またはMPEG1 audio layer 3の規格に準拠したエンコード処理を実行する。エンコーダ112はまた、エンコードし

て得られたPCM圧縮音楽データを、メディアムLAN102を介して、サーバ113に、例えば、ファイル転送ベース（（ファイル転送プロトコル（ftp:file transfer protocol））で転送する。

【0044】

サーバ113は、PCM非圧縮音楽データを記録し、それをエンコーダ112に供給したり、エンコーダ112からのPCM圧縮音楽データを記録し、要求に応じて、音楽配信サービスの利用者（受信側）に配信する。

【0045】

このように、制御データが転送されるアラームLAN101を、音楽データが転送されるメディアムLAN102とは別に設けるようにすることで、エンコード処理および音楽データの配信処理がより効率的に行われる。

【0046】

図3は、エンコーダ112の構成例を示している。CPU（Central Processing Unit）121は、ROM（Read Only Memory）122又はハードディスク127に記憶されている、例えば、エンコード処理用のプログラムをRAM（Random Access Memory）123に展開し、それに従ってエンコード処理を実行する。RAM123には、CPU121が、例えば、エンコード処理を実行する上において必要なデータなどが適宜記憶される。また、RAM123には、必要なデータバッファ領域が確保される。

【0047】

ネットワークカード124は、アラームLAN101に接続され、アラームLAN101を介して、制御データを受信したり、送信する。

【0048】

ネットワークカード125は、メディアムLAN102に接続され、メディアムLAN102を介して、サーバ113から送信されてくるPCM非圧縮音楽データを受信したり、エンコードカード126によりエンコードされたPCM圧縮音楽データを、サーバ113に送信する。

【0049】

エンコードカード126は、ネットワークカード125により受信されたPCM

非圧縮音楽データを、ATRAC1に準拠したエンコード処理（ATRAC1対応エンコード処理）、またはMEPG1 audio layer 3に準拠したエンコード処理（MPEG1対応エンコード処理）する。

【0050】

ハードディスク127は、CPU121が実行するプログラムの他、ネットワークカード125により受信されたPCM非圧縮音楽データや、エンコードカード126によりエンコードされたPCM圧縮音楽データを記憶する。

【0051】

この例の場合、音楽データは、イーサネットであるメディアムLAN102上に転送されるので、PCM非圧縮音楽データは、リアルタイムでエンコードされない。サーバ113から供給されたPCM非圧縮音楽データは、はじめに、ハードディスク127に記憶され、所定のタイミングで、エンコードカード126に供給され、エンコードされる。その後、エンコードされたPCM圧縮音楽データは、ハードディスク127に記憶される。このように、ハードディスク127を一時バッファとして利用することより、PCM非圧縮音楽データは、適切にエンコードされる。

【0052】

インターフェース128は、ネットワークカード124、ネットワークカード125、エンコードカード126、ハードディスク127と、CPU121との間に配置され、インターフェース処理を実行する。

【0053】

なお、CPU121、ROM122、RAM123、およびインターフェース128は、1つのマザーボード120に実装されている。

【0054】

この音楽配信サービスシステムは、リアルタイムOS (Oprating System) に基づき制御される。リアルタイムOSは、複数のタスク（プロセス）を同時に動作させることが可能なマルチタスクOSである。非リアルタイムOSでは、所定の処理を実行している最中に、他のイベント、すなわち外部からの要求が発生したとしても、そのイベントハンドラが直ちには起動されない場合がある。これに対してリ

リアルタイムOSでは、予め設定された所定の時間内に、イベントハンドラが起動されることが保証されている。すなわち、図4に示されるように、クロックにより定められた所定の時間（サバイバルタイム）内において開始された処理は、そのサバイバルタイム内において終了される。

【0055】

図5は、リアルタイムOSの基本的な概念を模式的に表している。各プロセスは、生成されると、動作状態、待機状態、および動作可能状態の3つの状態のいずれかに順次遷移するように管理される。各プロセスは、フラグ管理され、動作状態から待機状態に遷移するとき、フラグ待ちを指示するコマンドwaitFlagが発行され、待機状態から動作可能状態へと遷移するとき、フラグに値をセットするコマンドsetFlagが発行される。

【0056】

同時に複数のプロセスが存在可能であり、複数のプロセスのそれぞれには、優先順位が割り当てられる。リアルタイムOSにより、動作可能状態にあるプロセスの中で最も優先順位の高いプロセスに実行権が渡される。

【0057】

リアルタイムOSでは、このようにイベントの発生によって、動作状態にあるプロセスを高速に切り換えることができるようになされている。

【0058】

図6は、エンコーダ112の機能的構成例（エンコード処理用のプログラムの構成例）を示している。このエンコード処理用のプログラムは、カプセルマネージャ151、エンコード制御アプリケーション152、およびメインアプリケーション153の3つのプロセス（実施可能なプログラムとそれを実行するために必要なデータ領域を有している）の他、所定のプロセスがブロック化（カプセル化）された制御データ入出力カプセル154、音楽データ入出力カプセル155、およびエンコード処理カプセル156の3つのカプセル（図中、2重線の枠で示されている要素）から構成されている。なお、図中、実線の枠で示されている要素は、プロセスである。

【0059】

カプセルマネージャ151は、メインアプリケーション153を介して、エンコード制御アプリケーション152、制御データ入出力カプセル154、音楽データ入出力カプセル155、およびエンコード処理カプセル156を制御する。

【0060】

エンコード制御アプリケーション152は、メインアプリケーション153と通信して、例えば、カプセルマネージャ151からの指令を受信し、それに基づいて制御データ入出力カプセル154を制御する。メインアプリケーション153は、カプセルマネージャ151と直接通信し、カプセルマネージャ151からの指令に基づいて音楽データ入出力カプセル155およびエンコード処理カプセル156を制御する。

【0061】

制御データ入出力カプセル154は、エンコード制御マネージャ161により管理される。そのエンコード制御マネージャ161は、エンコード制御アプリケーション152からの指令により、ネットワークカード入力I/Fプロセス162（以下、ネットワークカード入力I/Fと略称する。他のプロセスについても同様である）、ネットワークカード出力I/F163、およびネットワークカードドライバ（プロセス）164を生成したり、消去する。

【0062】

ネットワークカード入力I/F162は、ネットワークカード124、およびネットワークカードドライバ164を介して、端末111の制御プロセスWからの制御データ（コマンド）を受信し、エンコード制御マネージャ161に出力する。ネットワークカード出力I/F163は、エンコード制御アプリケーション152とエンコード制御マネージャ161を介して、メインアプリケーション153からの制御データ（メッセージ）を受信し、ネットワークカードドライバ164に出力する。

【0063】

ネットワークカードドライバ164は、ネットワークカード124に対して入出力インターフェース処理を実行し、アラームLAN101上に転送される制御データ（コマンド）を受信し、ネットワークカード入力I/F162に出力したり、

ネットワークカード出力I/F163から、制御データ（メッセージ）を受信し、ネットワークカード124に出力する。

【0064】

音楽データ入出力カプセル155は、データ入出力マネージャ171により管理される。このデータ入出力マネージャ171は、メインアプリケーション153からの指令により、ネットワークカード入出力I/F172（プロセス）およびネットワークカードドライバ173（プロセス）を生成したり、消去する。

【0065】

ネットワークカード入出力I/F172は、ネットワークカード125とネットワークカードドライバ173を介して、サーバ113からPCM非圧縮音楽データを受信し、データ入出力マネージャ171に出力したり、データ入出力マネージャ171から、PCM圧縮音楽データを受信し、ネットワークカードドライバ173に出力する。

【0066】

ネットワークカードドライバ173は、ネットワークカード125に対して入出力インターフェース処理を実行し、メディアムLAN102上に転送されたPCM非圧縮音楽データを受信し、ネットワークカード出力I/F172に出力する。ネットワークカードドライバ173はまた、ネットワークカード入出力I/F172から、PCM圧縮音楽データを受信し、ネットワークカード125に出力する。

【0067】

エンコード処理カプセル156は、エンコード処理マネージャ181により管理される。このエンコード処理マネージャ181は、メインアプリケーション153からの指令により、エンコードエンジン入出力I/F182、エンコードエンジン入出力I/F183、エンコードエンジン184、エンコードカード入出力I/F185、エンコードカード入出力I/F186、およびエンコードカードドライバ187（いずれもプロセス）を生成したり、消去する。エンコード処理マネージャ181はまた、ソフトウェア（エンコードエンジン184）によるエンコードと、ハードウェア（エンコードカード126）によるエンコードの識別処理、並びに、ATRAC1対応エンコード処理とMPEG1対応エンコード処理の識別処理を行

う。

【0068】

エンコードエンジン入出力I/F182は、エンコード処理マネージャ181から、ATRAC1対応エンコード処理されるPCM非圧縮音楽データを受信し、エンコードエンジン184に出力する。エンコードエンジン入出力I/F182はまた、エンコードエンジン184から、ATRAC1対応エンコード処理されたPCM圧縮音楽データを受信し、エンコード処理マネージャ181に出力する。

【0069】

エンコードエンジン入出力I/F183は、エンコード処理マネージャ181から、MPEG1対応エンコード処理されるPCM非圧縮音楽データを受信し、エンコードエンジン184に出力する。エンコードエンジン入出力I/F183はまた、エンコードエンジン184から、MPEG1対応エンコード処理されたPCM圧縮音楽データを受信し、エンコード処理マネージャ181に出力する。

【0070】

エンコードエンジン184は、ソフトウェアによるATRAC1対応エンコード処理、またはソフトウェアによるMPEG1対応エンコード処理を、エンコードエンジン入出力I/F182、183から供給された制御データに基づき実行する。

【0071】

エンコードカード入出力I/F185は、エンコード処理マネージャ181から、ATRAC1対応エンコード処理されるPCM非圧縮音楽データを受信し、エンコードカードドライバ187に出力する。エンコードカード入出力I/F185はまた、エンコードカードドライバ187から、ATRAC1対応エンコード処理されたPCM圧縮音楽データを受信し、エンコード処理マネージャ181に出力する。

【0072】

エンコードカード入出力I/F186は、エンコード処理マネージャ181から、MPEG1対応エンコード処理されるPCM非圧縮音楽データを受信し、エンコードカードドライバ187に出力する。エンコードカード入出力I/F186はまた、エンコードカードドライバ187から、MPEG1対応エンコード処理されたPCM圧縮音楽データを受信し、エンコード処理マネージャ181に出力する。

【0073】

エンコードカードドライバ187は、エンコードカード126に対して入出力インターフェース処理を実行し、エンコードカード入出力I/F185、186から、PCM非圧縮音楽データを受信し、エンコードカード126に出力したり、エンコードカード126から、エンコード処理されたPCM圧縮音楽データを受信し、エンコードカード入出力I/F185、186に出力する。ハードウェアとしてのエンコードカード126は、ATRAC1対応エンコード処理するとき、対応する機能を有するものに取り替えられ、MPEG1対応エンコード処理するとき、対応する機能を有するものに取り替えられる。

【0074】

次に、図7のフローチャートを参照して、ATRAC1対応エンコード処理を実行する場合における各プロセスの全体的な処理について説明する。

【0075】

なお、電源がオンされると、リアルタイムOSがカプセルマネージャ151を生成すると共に、カプセルマネージャ151は、メインアプリケーション153に、音楽データ入出力カプセル155とエンコード処理カプセル156を生成させるとともに、エンコード制御アプリケーション152に、制御データ入出力カプセル154を生成させる。

【0076】

端末111の制御プロセスWから、アラームLAN101を介してエンコーダ112のネットワークカード124にATRAC1対応エンコード処理が指令されると、この指令は、制御データ入出力カプセル154のネットワークカードドライバ164からネットワークカード入力I/F162を介してエンコード制御マネージャ161に供給される。エンコード制御マネージャ161は、この指令を受けると、ステップS1において、エンコード制御アプリケーション152に対して、ATRAC1対応エンコード処理の開始を要求する。

【0077】

エンコード制御アプリケーション152は、ステップS11において、エンコード制御マネージャ161からの要求を受け取る。この要求には、エンコードの

種類とエンコードを実行する処理部を特定するコード情報が含まれている。この例の場合、エンコードの種類は、ATRAC1 対応エンコード処理とされ、エンコードの実行処理部は、ハードウェア（エンコードカード126）とされる。

【0078】

エンコード制御アプリケーション152は、エンコード制御マネージャ161からの要求を受け取ると、ステップS12において、その要求に基づいて、ATRAC1 対応エンコード処理の許可をメインアプリケーション153に要求する。

【0079】

メインアプリケーション153は、ステップS31において、エンコード制御アプリケーション152からの要求を受け取ると、その要求に対応する処理を実行できるか否かを判定し、実行できない場合には、その旨をエンコード制御アプリケーション152に応答する。エンコード制御アプリケーション152は、この応答結果を受信した場合、対応する応答を、さらにエンコード制御マネージャ161に通知する。エンコード制御マネージャ161は、この通知に対応する応答を、さらにエンコード制御マネージャ161、ネットワークカード出力I/F163、ネットワークカードドライバ164、ネットワークカード124を介して、端末111の制御プロセスWに通知する。

【0080】

図7の例の場合、メインアプリケーション153は、ATRAC1 対応エンコード処理を実行できると判定し、対応する許可応答を、ステップS32において、エンコード制御アプリケーション152に通知する。

【0081】

エンコード制御アプリケーション152は、ステップS13において、メインアプリケーション153からの許可応答を受信すると、ステップS14において、エンコード制御マネージャ161に、ATRAC1 対応エンコード処理の開始を示す応答を出力する。エンコード制御マネージャ161は、この応答を、ステップS2において受信する。エンコード制御マネージャ161は、この応答に対応する通知を、上述したようにして、さらに制御プロセスWに通知する。

【0082】

メインアプリケーション 1 5 3 は、ステップ S 3 2 において、ATRAC 1 対応エンコード処理の許可応答をエンコード制御アプリケーション 1 5 2 に出力した後、ステップ S 3 3 において、音楽データ入出力カプセル 1 5 5 のデータ入出力マネージャ 1 7 1 に、エンコード処理の対象とされる基データの取得を要求する（基データ取得要求を出力する）。この基データ取得要求には、その基データが記憶されている記憶部のアドレスの他、取得するデータ量等も含まれている。

【 0 0 8 3 】

ステップ S 8 1 において、データ入出力マネージャ 1 7 1 は、メインアプリケーション 1 5 3 からの要求を受け取ると、ステップ S 8 2 において、その要求に基づいて、エンコードする基データを、指定されたアドレスから取得する処理を実行する。

【 0 0 8 4 】

例えば、基データがハードディスク 1 2 7 に記憶されている場合、データ入出力マネージャ 1 7 1 は、ネットワークカード入出力 I/F 1 7 2、ネットワークカードドライバ 1 7 3、ネットワークカード 1 2 5 を介してハードディスク 1 2 7 にアクセスし、指定されたアドレスから、指定された量の PCM 非圧縮音楽データを取得する。

【 0 0 8 5 】

なお、データ入出力マネージャ 1 7 1 は、基データが図示せぬ CD-R 等に記録されている場合には、そこにアクセスし、基データを取得する。また、データ入出力マネージャ 1 7 1 は、基データがサーバ 1 1 3 に記録されている場合には、アラーム LAN 1 0 1 を介してサーバ 1 1 3 にアクセスし、ミディウム LAN 1 0 2 を介してサーバ 1 1 3 から基データの転送を受ける。取得された PCM 非圧縮音楽データは、RAM 1 2 3 のバッファ領域に、一時的に格納される。

【 0 0 8 6 】

このようにして、PCM 非圧縮音楽データが取得されると、ステップ S 8 3 において、データ入出力マネージャ 1 7 1 は、メインアプリケーション 1 5 3 に対して、基データが取得されたことを表す基データ取得応答を送信する。

【 0 0 8 7 】

メインアプリケーション 1 5 3 は、ステップ S 3 4 において、データ入出力マネージャ 1 7 1 からの基データ取得応答を受信すると、ステップ S 3 5 において、エンコード処理マネージャ 1 8 1 にエンコード開始を要求する。この要求には、エンコードの種類が ATRAC 1 対応エンコード処理であること、並びにエンコードの実行処理部がエンコードカード 1 2 6 であることを表す情報が含まれている。

【 0 0 8 8 】

エンコード処理マネージャ 1 8 1 は、ステップ S 6 1 において、メインアプリケーション 1 5 3 からのエンコード開始要求を受信すると、この要求に対応する処理を実行できるか否かを判定し、実行できない場合には、その旨を表す応答をメインアプリケーション 1 5 3 に出力する。図 7 の例の場合、エンコード処理マネージャ 1 8 1 は、エンコードが実行できると判定し、ステップ S 6 2 において、エンコードを開始することを表す応答をメインアプリケーション 1 5 3 に出力する。

【 0 0 8 9 】

メインアプリケーション 1 5 3 は、エンコード処理マネージャ 1 8 1 からのエンコード開始応答を、ステップ S 3 6 において受信する。これにより、メインアプリケーション 1 5 3 は、ステップ S 3 5 の要求に対応する処理が、エンコード処理マネージャ 1 8 1 において受け入れられたことを知ることができる。

【 0 0 9 0 】

エンコード処理マネージャ 1 8 1 は、ステップ S 6 2 において、メインアプリケーション 1 5 3 にエンコード開始応答を出力した後、ステップ S 6 3 においてエンコード処理を実行させる。具体的には、エンコード処理マネージャ 1 8 1 は、メインアプリケーション 1 5 3 からのエンコード開始要求に基づいて、エンコードカード入出力 I/F 1 8 5、エンコードカードドライバ 1 8 7 を介してエンコードカード 1 2 6 に、ATRAC 1 対応エンコード処理を要求する。この要求には、エンコード対象とされる PCM 非圧縮音楽データが保持されている RAM 1 2 3 のバッファのアドレスが含まれている。エンコードカード 1 2 6 は、この要求に基づいて、RAM 1 2 3 のバッファ領域に記憶されている PCM 非圧縮音楽データを読み出し

、ATRAC 1 対応エンコード処理を実行する。エンコードされたPCM圧縮音楽データは、RAM 1 2 3 のバッファ領域に記憶される。

【 0 0 9 1 】

以上のようにして、エンコードが終了すると、ステップ S 6 4 において、エンコード処理マネージャ 1 8 1 は、エンコードが終了したことを表すエンコード終了通知を、メインアプリケーション 1 5 3 に出力する。

【 0 0 9 2 】

メインアプリケーション 1 5 3 は、ステップ S 3 7 において、エンコード処理マネージャ 1 8 1 が出力したエンコード終了通知を受信すると、ステップ S 3 8 において、これに対応するエンコード終了応答を、エンコード処理マネージャ 1 8 1 に通知する。

【 0 0 9 3 】

エンコード処理マネージャ 1 8 1 は、ステップ S 6 5 において、メインアプリケーション 1 5 3 より出力されたエンコード終了応答を受信する。これにより、エンコード処理マネージャ 1 8 1 は、エンコード終了が、メインアプリケーション 1 5 3 に通知されたことを確認することができる。

【 0 0 9 4 】

メインアプリケーション 1 5 3 は、ステップ S 3 8 において、エンコード処理マネージャ 1 8 1 にエンコード終了応答を出力した後、ステップ S 3 9 において、データ入出力マネージャ 1 7 1 に対して、エンコードした結果生成されたPCM圧縮音楽データを保存することを要求する生成データ保存要求を出力する。この要求には、データを保存する媒体のアドレスが含まれている。例えば、ハードディスク 1 2 7 に、このエンコード後のPCM圧縮音楽データを保存する場合、ハードディスク 1 2 7 のアドレスが含まれている。

【 0 0 9 5 】

データ入出力マネージャ 1 7 1 は、ステップ S 8 4 において、メインアプリケーション 1 5 3 からの生成データ保存要求を受け取ると、ステップ S 8 5 において、その要求に基づいて、受信したデータを保存する処理を実行する。具体的には、データ入出力マネージャ 1 7 1 は、例えば、ハードディスク 1 2 7 への保存

が要求された場合、ネットワークカード入出力I/F172、ネットワークカードドライブ173、ネットワークカード125を介して、ハードディスク127にアクセスし、指定されたアドレスにエンコードされたPCM圧縮音楽データを記録させる。

【0096】

保存先がCD-Rとして指定されている場合には、エンコード後のPCM圧縮音楽データは、CD-Rの指定されたアドレスに保存される。また、保存先がサーバ113に指定されている場合、エンコード後のPCM圧縮音楽データは、メディアムLAN102を介してサーバ113に供給され、保存される。

【0097】

以上のようにして保存が完了したとき、ステップS86において、データ入出力マネージャ171は、データが保存されたことを表す生成データ保存応答を、メインアプリケーション153に出力する。

【0098】

メインアプリケーション153は、ステップS40において、データ入出力マネージャ171からの生成データ保存応答を受信すると、ステップS41において、ATRAC1対応エンコード処理が終了したことを表す終了通知を、エンコード制御アプリケーション152に出力する。

【0099】

エンコード制御アプリケーション152は、ステップS15において、メインアプリケーション153からの通知を受信すると、ステップS16において、エンコード制御マネージャ161に、ATRAC1対応エンコード処理が終了したことを通知する。

【0100】

エンコード処理マネージャ161は、ステップS3において、エンコード制御アプリケーション152からの終了通知を受信すると、ステップS4において、この通知に対応する応答を、エンコード制御アプリケーション152に出力する。

【0101】

エンコード制御アプリケーション 1 5 2 は、ステップ S 1 7 において、エンコード制御マネージャ 1 6 1 からの応答を受信すると、ステップ S 1 8 において、メインアプリケーション 1 5 3 に対して ATRAC 1 対応処理終了応答を送信する。

【 0 1 0 2 】

メインアプリケーション 1 5 3 は、ステップ S 4 2 において、エンコード制御アプリケーション 1 5 2 からの終了応答を受信する。

【 0 1 0 3 】

図 8 は、MPEG 1 対応エンコード処理が実行される場合の処理を表している。図 8 のステップ S 1 0 1 乃至ステップ S 1 4 2 は、図 7 の ATRAC 1 対応エンコード処理を実行する場合における、ステップ S 1 乃至ステップ S 4 2 における処理と、基本的に同様の処理である。ただし、エンコードの種類が、ATRAC 1 対応エンコード処理ではなく、MPEG 1 対応エンコード処理とされている点異なる。従って、図 7 のステップ S 8 2 に対応する図 8 のステップ S 1 8 2 における基データ取得処理では、ATRAC 1 対応処理のフォーマットに基づくデータの送出量や送出タイミングが、MPEG 1 対応エンコード処理のフォーマットに基づくものに変更される。

【 0 1 0 4 】

また、図 7 のステップ S 6 3 の ATRAC 1 対応エンコード処理に対応する図 8 のステップ S 1 6 3 における処理では、ATRAC 1 対応エンコード処理ではなく、MPEG 1 対応エンコード処理が実行される。当然のことながら、ATRAC 1 と MPEG 1 とは、それぞれ異なるエンコード方式であるため、エンコードのデータ量の単位、送出タイミング、誤り訂正符号等も異なったものとなる。

【 0 1 0 5 】

このことは、図 7 のステップ S 8 5 に対応する図 8 のステップ S 1 8 5 における処理でも同様である。

【 0 1 0 6 】

しかしながら、これらのエンコードの方式が異なることに起因する異なる処理は、全てエンコード処理カプセル 1 5 6 内のエンコード処理マネージャ 1 5 1 を中心とする各プロセスや、音楽データ入出力カプセル 1 5 5 のデータ入出力マネ

ージャ 1 7 1 を中心とする各プロセスが実行する。従って、メインアプリケーション 1 5 3、エンコード制御アプリケーション 1 5 2 等は、ATRAC 1 対応エンコード処理と MPEG 1 対応エンコード処理の違いを微細に認識している必要がない。

【 0 1 0 7 】

すなわち、メインアプリケーション 1 5 3 から見て、音楽データ入出力カプセル 1 5 5 のデータ入出力マネージャ 1 7 1 により実行されるデータ取得処理および保存処理、並びにエンコード処理カプセル 1 5 6 のエンコード処理マネージャ 1 8 1 により実行されるエンコード処理は、その詳細が隠ぺいされる。

【 0 1 0 8 】

例えば、データ入出力マネージャ 1 7 1 は、ハードディスク 1 2 7 からデータを取得する際、ATA (AD(Advanced Technology) Attachment) コマンドを用いるが、ネットワーク上のサーバ 1 1 3 からデータを取得する際は、TCP/IP (Transmission Control Protocol/Internet Protocol) 上の FTP (File Transfer Protocol)、または HTTP (Hyper Text Transport Protocol) を用いる。

【 0 1 0 9 】

また、CD-R からデータを取得する場合には、SCSI (Small Computer System Interface) インターフェースが用いられ、CD-R を用いてデータが取得される場合、ATAPI (AT Attachment Packet Interface) が用いられる。

【 0 1 1 0 】

また、エンコード処理は、ATRAC 形式のエンコードを高速に行う際、エンコードカードドライバ 1 8 7 のうち、ATRAC エンコーダドライバを介して処理が行われるが、MPEG フォーマットのエンコードを高速に行う場合には、エンコードカードドライバ 1 8 7 のうち、MPEG エンコーダドライバを介して処理が行われる。

【 0 1 1 1 】

このように、エンコード処理カプセル 1 5 6 の内部においては、様々なハードウェアドライバプロトコルを用いて処理が行われるが、メインアプリケーション 1 5 3 は、データ取得要求、エンコード開始要求、およびデータ保存要求という簡単な 3 つのシーケンスだけを有していれば良い。

【 0 1 1 2 】

従って、エンコード処理カプセル156や音楽データ入出力カプセル155は、これらのプロトコルやデータ取得方法を、メインアプリケーション153に対して隠ぺいするものとなる。

【0113】

結果的に、メインアプリケーション153やエンコード制御アプリケーション152は、ただ単に、エンコードの種類や、エンコードを実行する処理が、ソフトウェアであるのかハードウェアであるのかを識別するだけでよい。従って、メインアプリケーション153やエンコード制御アプリケーション152は、基本的に、エンコードの種類が変更されたり、拡張されても変更する必要がないか、変更するにしても、わずかで済む。

【0114】

エンコーダ112のエンコード処理用のプログラムは、以上のように構成されるが、図6に示されるプロセス（図中、実線の枠で示されている要素）のそれぞれは、図9に示す状態遷移図に従ってステート（状態）を遷移し、所定の処理を実行する。

【0115】

なお、この図9の各ステートは、図5における動作状態を構成するものである。

【0116】

図9の状態遷移図におけるステートは、イグゼキュート (EXECUTE)、イニシャライズ0 (INITIALIZE0)、イニシャライズ1 (INITIALIZE1)、イニシャライズ2 (INITIALIZE2)、ターミネート1 (TERMINATE1)、ターミネート0 (TERMINATE0)、エグジット (EXIT)、レシーブ (RECEIVE)、リプライ (REPLY)、マネージ (MANAGE)、SEND (SEND)、およびレディ (READY) の12種類のステートにより構成されている。

【0117】

次に、図9の状態遷移図の正常時における場合の状態遷移について、図10のフローチャートを参照して説明する。

【0118】

電源がオンされ、リアルタイムOSが立ち上げられてプロセスが生成されると、ステップS 2 0 1において、プロセスは、イグゼキュートステート (EXECUTE) Aとなる。

【 0 1 1 9 】

次に、ステップS 2 0 2において、プロセスは、イニシャライズ0ステート (INITIALIZE0) Bに遷移する。このイニシャライズ0ステート (INITIALIZE0) Bにおいては、カプセル (モジュール) 登録処理が実行される。具体的には、主に、ハードウェアとのデータの授受に用いられる第1のバッファが確保され、さらに、その第1のバッファが初期化され (例えば、0が埋め込まれ)、デバイスがオープンになり、そしてデバイスが正常であるか否かがチェックされる。この処理は、電源オン時に、1回だけ必要な処理である。

【 0 1 2 0 】

次に、ステップS 2 0 3において、プロセスは、イニシャライズ0ステート (INITIALIZE0) Bからイニシャライズ1ステート (INITIALIZE1) Cに遷移し、カプセル用のリソース確保処理を実行する。この処理が実行されることより、主に、メインアプリケーション153との間のデータの授受に用いられる第2のバッファが確保され、さらに、その第2のバッファが初期化され、そしてレジスタが初期化される。

【 0 1 2 1 】

ステップS 2 0 4において、プロセスは、イニシャライズ1ステート (INITIALIZE1) Cから、イニシャライズ2ステート (INITIALIZE2) Dに遷移し、デバイスを含むカプセルの初期化処理を実行する。この初期化処理が実行されることより、第1のバッファが再び初期化される (このとき、確保処理は行われない)

【 0 1 2 2 】

次に、ステップS 2 0 5において、プロセスは、イニシャライズ2ステート (INITIALIZE2) Dからレディステート (READY) Eに遷移する。

【 0 1 2 3 】

ステップS 2 0 5において、各プロセスは、図9におけるステートE, F, G

、H、I、J、K、L、Mのいずれかのステートに遷移する。すなわち、ステップS206において、各プロセスは、終了要求の処理要求メッセージを受信したか否かを判定し、終了要求以外の処理要求メッセージを受信した場合には、ステップS205に戻り、対応するステートに遷移する。

【0124】

ステートE乃至ステートMのいずれかのステートにあるプロセスは、ステップS206において、終了要求の処理要求メッセージを受信したと判定した場合、ステップS207に進み、ターミネート1ステート (TERMINATE1) Nに遷移し、ここで、第2のバッファの開放処理が行われる。次に、ステップS208に進み、プロセスは、ターミネート0ステート (TERMINATE0) Oに遷移する。このステートにおいては、第1のバッファが開放されるとともに、デバイスがクローズされる。

【0125】

さらに、ステップS209に進み、プロセスは、エグジットステート (EXIT) Pに遷移し、リアルタイムOSが終了され、電源がオフされた状態に移行する。

【0126】

各プロセスは、他のプロセスと、プロセス間通信用メッセージを授受し、所定のステートから、図9において矢印で示される他のステートに遷移する。

【0127】

図11は、このプロセス間通信用メッセージのフォーマットを表している。プロセス間通信用メッセージは、図11(A)に示すように、ヘッダと拡張部分からなり、それらには、FTP (File Transfer Protocol)のフォーマットで各データが記載されている。ヘッダの” unsigned short type;” には、図11(C)に示すように、メッセージタイプ (以下、MSGT(Message Type)と記述する)、ファンクションタイプ (以下、FNCT(Function Type)と記述する)、メッセージナンバー (以下、MSGN(Message number)と記述する)、そしてファンクションナンバー (以下、FNCN(Function Number)と記述する) が設けられている。

【0128】

MSGTには、下記に示すように、要求するインタラプト、センド、レシーブ、ま

たはリプライの各処理に対応して割り当てられている 0 0 乃至 1 1 のデータが格納される。

<処理>	<データ>
インタラプト (INT (INTERRUPT))	0 0 B
センド (SND (SEND))	0 1 B
レシーブ (RCV (RECEIVE))	1 0 B
リプライ (RPY (REPLY))	1 1 B

【0 1 2 9】

MSGNには、下記に示すように、要求する処理に対応して割り当てられている所定のデータが格納される。

<処理>	<データ>
クワイト (QUT (QUIT))	0 0 1 B
リセット (RST (RESET))	0 1 0 B
ターミネート (TRM (TERMINATE))	0 1 1 B
リクエスト (REQ (REQUEST))	1 0 0 B
ノーティファイ (NTF (NOTIFY))	1 0 1 B
リフューズ (RFS (REFUSE))	1 1 0 B
アクノリッジメント (ACK (ACKNOWLEDGEMENT))	1 1 1 B

【0 1 3 0】

FNCTおよびFNCNには、要求する処理に対応して、それぞれ下記の事項を示すデータが格納される。

<FNCT>	<FNCN>
エンコード	ATRAC 1 対応エンコード処理 MPEG 1 対応エンコード処理
ファイル転送	put get

【0 1 3 1】

ヘッダの " pid_t src_pid" には、プロセス間通信用メッセージの送り元 (source) のプロセスのIDが、 " pid_t dst_pid" には、送り先 (destination) のプロセ

スのIDが、それぞれ記載される。

【0132】

プロセス間通信用メッセージの拡張部分には、例えば、エンコードされるPCM非圧縮音楽データが格納されている場所（例えば、ハードディスク127のメモリアドレス）や、エンコードされたPCM圧縮音楽データを格納する場所などが記載されている。

【0133】

図12のプロセス間通信用メッセージは、図11のプロセス間通信用メッセージのヘッダのみから構成されている。

【0134】

次に、図10のステップS205、S206における図9のステートE乃至ステートMの間における状態遷移を説明する。ここでは、端末111（制御プロセスW）からの要求に基づいて、メインアプリケーション153が、エンコード処理マネージャ181に、ATRAC1対応エンコード処理を要求する場合を例として説明する。

【0135】

なお、各プロセスは、図13に示されるように、ステップS221において、処理待ちの状態から、ステップS222に進み、所定の処理を実行し、その処理が完了すると、再び、ステップS221の処理待ちの状態に戻るようプログラムされている。従って、これから説明するステートの遷移は、各プロセスが規定しているものであり、いずれの方向のステートに遷移するかがステートそのものに規定されているわけではない。換言すれば、各プロセスは、図9に矢印で示されるルートに従って遷移することが予め規則付けられている。

【0136】

はじめに、メインアプリケーション153の状態遷移を、図14のフローチャートを参照して説明する。なお、メインアプリケーション153のステートは、図10のステップS201乃至S205の処理で、すでに、レディステート（READY）Eに遷移しているものとする。

【0137】

ステップ S 3 1 1 において、メインアプリケーション 1 5 3 は、MSGT に SND が、MSGN に REQ が、FNCT にエンコードを示すデータが、そして FNCN に ATRAC 1 対応エンコード処理を示すデータが、それぞれ設定されているヘッダを含むプロセス間通信用メッセージ（以下、このように MSGT に SND が、そして MSGN に REQ が設定されているプロセス間通信用メッセージを、(SND, REQ) メッセージと記載する。他のプロセス間通信用メッセージの場合も同様とする）を、エンコード処理マネージャ 1 8 1 に送信し、レディステート (READY) E からセンドステート (SEND) J に遷移する。

【 0 1 3 8 】

なお、この (SND, REQ) メッセージのヘッダには、送り元のメインアプリケーション 1 5 3 の ID が設定された "pid_t src_pid"、および送り先であるエンコード処理マネージャ 1 8 1 の ID が設定された "pid_t dst_pid" が含まれている。また (SND, REQ) メッセージは、図 1 1 に示す構成を有しており、その拡張部分には、エンコードされる PCM 非圧縮音楽データが格納されているハードディスク 1 2 7 のメモリアドレス、エンコードされた後の PCM 圧縮音楽データが格納されるハードディスク 1 2 7 のメモリアドレスが記述されている。

【 0 1 3 9 】

また、図 9 において、括弧を付加して示すメッセージは、受信メッセージを表し、括弧を付加しないメッセージは送信メッセージを表す。

【 0 1 4 0 】

メインアプリケーション 1 5 3 はまた、このとき、内蔵するタイマー t をスタートさせる。

【 0 1 4 1 】

次に、ステップ S 3 1 2 において、メインアプリケーション 1 5 3 は、ステップ S 3 1 1 で送信した (SND, REQ) メッセージに応答するプロセス間通信用メッセージ (MSGT に RPY が設定されているメッセージで、以下、このようなメッセージを (RPY, XXX) メッセージと記述する) が、エンコード処理マネージャ 1 8 1 から受信されるまで待機し、受信されたと判定した場合、ステップ S 3 1 3 に進む。なお、(RPY, XXX) メッセージは、図 1 2 に示した構成を有している。なお

、(RPY, XXX) メッセージは、図 12 に示した構成を有している。すなわち、ヘッダのみから構成されている。

【0142】

ステップ S 313 において、メインアプリケーション 153 は、ステップ S 312 で受信されたと判定された (RPY, XXX) メッセージが、(RPY, ACK) メッセージ (MSGT に RPY が、MSGN に ACK が、それぞれ設定されているメッセージ) であるか否かを判定し、(RPY, ACK) メッセージであると判定した場合、ステップ S 314 に進む。この例の場合のように、エンコード処理が要求されているとき、エンコード処理マネージャ 181 は、エンコード処理が可能であるとき、(RPY, ACK) メッセージをメインアプリケーション 153 に送信してくる。

【0143】

ステップ S 314 において、メインアプリケーション 153 は、センドステート (SEND) J からマネージステート (MANAGE) K に遷移し、所定の処理を実行する。ただし、いまの場合、メインアプリケーション 153 は、特に処理を行わない。このマネージステート (MANAGE) K は、各プロセスに、必要に応じて所定の処理を実行させることができるように用意されている。

【0144】

次に、ステップ S 315 において、メインアプリケーション 153 は、エンコード終了待ちの (RCV, REQ) メッセージをエンコード処理マネージャ 181 に送信して、マネージステート (MANAGE) K からレシーブステート (RECEIVE) L に遷移する。ステップ S 316 において、メインアプリケーション 153 は、エンコード処理マネージャ 181 から、エンコード終了待ちメッセージの受領を通知する (SND, NTF) メッセージが受信されるまで待機し、受信されたと判定した場合、ステップ S 317 に進む。

【0145】

ステップ S 317 において、メインアプリケーション 153 は、レシーブステート (RECEIVE) L からリプライステート (REPLY) M に遷移し、ステップ S 316 で受信したと判定された (SND, NTF) メッセージに応答するための (RPY, NTF) メッセージをエンコード処理マネージャ 181 に送信する。その後、ステップ

S 3 1 8に進み、メインアプリケーション153は、リプライステート (REPLY) Mからレディステート (READY) Eに遷移し、処理は終了する。

【0146】

エンコード処理マネージャ181は、メインアプリケーション153からエンコード要求の (SND, REQ) メッセージを受信した場合、何らかの理由によりエンコードを実行することができないとき、エンコード不可を通知する (RPY, NTF) メッセージを送信してくる。そこで、この場合、ステップS 3 1 3において、受信された (RPY, XXX) メッセージが (RPY, ACK) メッセージではないと判定され、ステップS 3 1 4乃至ステップS 3 1 7の処理はスキップされ、ステップS 3 1 8に進み、メインアプリケーション153は、センドステート (SEND) Jからレディステート (READY) Eに遷移し、処理は終了する。

【0147】

なお、メインアプリケーション153は、センドステート (SEND) J、マネージステート (MANAGE) K、およびレシーブステート (RECEIVE) Lでは、ステップS 3 1 1でスタートさせたタイマー t の計測時間が所定の時間を超えているかを判定し、超えていないと判定した場合、上述した処理を実行するが、タイマー t の計測時間が所定の時間を超えていると判定した場合 (いわゆるタイムアウトの場合)、レディステート (READY) Eに遷移し、処理を終了させる。

【0148】

次に、この例の場合のエンコード処理マネージャ181の状態遷移を、図15のフローチャートを参照して説明する。ステップS 3 2 1において、エンコード処理マネージャ181は、エンコード要求待ちの (RCV, REQ) メッセージをメインアプリケーション153に送信し、レディステート (READY) Eからレシーブステート (RECEIVE) Fに遷移する。エンコード処理マネージャ181はまた、このとき、内蔵するタイマー t をスタートさせる。

【0149】

ステップS 3 2 2において、メインアプリケーション153から送信されてきた、エンコードを要求する (SND, REQ) メッセージ (図14のステップS 3 1 1の処理で送信されたメッセージ) が受信されると、エンコード処理マネージャ1

81は、ステップS323において、いま要求されたエンコードが処理可能であるか否かを判定する。エンコードが処理可能である場合には、ステップS324に進み、エンコード処理マネージャ181は、レシーブステート (RECEIVE) Fからリプライステート (REPLY) Gに遷移する。

【0150】

ステップS324において、エンコード処理マネージャ181は、メインアプリケーション153からのエンコード要求の (SND, REQ) メッセージに対応するエンコード受付応答の (RPY, ACK) メッセージを、メインアプリケーション153に送信する。そして、ステートは、マネージステート (MANAGE) Hに遷移する。

【0151】

次に、ステップS325において、エンコード処理マネージャ181は、マネージステート (MANAGE) Hにおいて、エンコードカード入出力I/F185とエンコードカードドライバ187を介してエンコードカード126を制御し、要求されたATRAC1対応エンコード処理を実行させる。

【0152】

カプセルマネージャ151は、制御データ入出力カプセル154、音楽データ入出力カプセル155、およびエンコード処理カプセル156の各カプセル、またはそのカプセルのプロセスのそれぞれに対する、優先順位およびスケジュールアルゴリズムを決定する。

【0153】

例えば、PCM非圧縮音楽データのエンコードが開始されるとき、エンコード処理カプセル156には、制御データ入出力カプセル154および音楽データ入出力カプセル155よりも高い優先順位が与えられる。また、ATRAC1対応エンコード処理が実行される場合、エンコード処理カプセル156のエンコード処理マネージャ181には、最も高い優先順位が与えられ、エンコードエンジン入出力I/F182、エンコードエンジン184、エンコードカード入出力I/F185、およびエンコードカードドライバ187には、それに続く優先順位が与えられ、そしてエンコードエンジン入出力I/F183およびエンコードカード入出力I/F18

6には、最も低い優先順位が与えられる。

【0154】

つまり、ATRAC1対応エンコード処理が実行される場合、ATRAC1対応エンコード処理に関する音楽データの入出力を、エンコードエンジン184との間でインターフェースするエンコードエンジン入出力I/F182、およびATRAC1対応エンコード処理に関する音楽データの入出力をエンコードカードドライバ187との間でインターフェースするエンコードカード入出力I/F185の優先順位は、MPEG1対応エンコード処理に関する音楽データの入出力をエンコードエンジン184との間でインターフェースするエンコードエンジン入出力I/F183、およびMPEG1対応エンコード処理に関する音楽データの入出力をエンコードカードドライバ187との間でインターフェースするエンコードカード入出力I/F186の優先順位に比べ、高く設定される。

【0155】

さらに、ATRAC1対応エンコード処理が実行される場合であっても、ソフトウェアによるエンコードではなく、ハードウェアによるエンコードが指令されているとき、エンコードエンジン入出力I/F182よりエンコードカード入出力I/F185の方が、優先順位が高く設定される。

【0156】

カプセルマネージャ151は、上述したようにして決定した、カプセルおよびプロセスの優先順位を、エンコード制御アプリケーション152およびメインアプリケーション153を介して、制御データ入出力カプセル154、音楽データ入出力カプセル155、およびエンコード処理カプセル156の各マネージャ（エンコード制御マネージャ161、データ入出力マネージャ171、およびエンコード処理マネージャ181）に供給する。そして、各マネージャは、通知されたプロセスの優先順位に基づいて、プロセスを生成し、処理を実行させる。

【0157】

例えば、ATRAC1対応エンコード処理が実行される場合、エンコード処理カプセル156のエンコード処理マネージャ181は、エンコードエンジン入出力I/F182、エンコードエンジン184、エンコードカード入出力I/F185、およ

びエンコードカードドライバ187を生成し、プロセス間通信用メッセージを供給する。生成されたエンコードエンジン入出力I/F182、エンコードエンジン184、エンコードカード入出力I/F185、およびエンコードカードドライバ187は、供給されたプロセス間通信用メッセージを、通信可能なプロセスと送受信し、エンコード処理を実行する。これにより、この例の場合、プロセス間通信用プロセスの拡張部分に設定されたアドレスに格納されているPCM非圧縮音楽データがハードディスク127から読み出され、ATRAC1対応エンコード処理が施され、ハードディスク127の、プロセス間通信用プロセスの拡張部分に設定されているアドレスに記憶される。

【0158】

なお、図15の処理の例の場合、ATRAC1対応エンコード処理をハードウェアにより実行するので、エンコードエンジン入出力I/F182、183、エンコードエンジン184、およびエンコードカード入出力I/F186は生成されない。これにより、それらに必要なリソースを確保する必要がなく、そのリソースを他の処理に利用することができる。

【0159】

図15の説明に戻り、ステップS325において、エンコード処理が完了したとき、ステップS326において、エンコード処理マネージャ181は、マネージステート (MANAGE) Hからセンドステート (SEND) Iに遷移し、エンコード終了を示す (SND, NTF) メッセージをメインアプリケーション153に送信する。

【0160】

ステップS327において、エンコード処理マネージャ181は、メインアプリケーション153から、送信された、エンコード終了待ち受領を通知する (RPY, NTF) メッセージ (図14のステップS317の処理で送信されたメッセージ) が受信されるまで待機し、受信されたと判定した場合、ステップS328に進み、センドステート (SEND) Iからレディステート (READY) Eに戻り、処理は終了する。

【0161】

マネージステート (MANAGE) H、およびセンドステート (SEND) Iにおいて、

エンコード処理マネージャ181は、ステップS321でスタートさせたタイマー t の計測時間が所定の時間を超えているか否かを判定し、超えていないと判定した場合、上述したように処理を実行するが、タイマー t の計測時間が所定の時間を超えていると判定した場合、レディステート (READY) Eに遷移し、処理を終了させる。

【0162】

以上のように、各プロセスが、プロセス間通信用メッセージの内容に基づいて、ステートを遷移するようにしたので、メインアプリケーション153が、エンコードするデータが格納されている場所、エンコードされたPCM圧縮音楽データが格納される場所、またはエンコード処理の種類をエンコード処理マネージャ181に通知するだけで、エンコード処理マネージャ181に所定のエンコード処理を実行させることができる。すなわち、メインアプリケーション153は、エンコード処理カプセル156内のエンコード処理マネージャ181以外のプロセスを制御したり（例えば、データ処理やバッファ処理を制御したり）、データ転送を制御する必要がない。従って、例えば、エンコードカード126が、より性能が高い他のエンコードカードと入れ替えられても、そのエンコードカードと、それに対応したエンコード処理カプセル156がともに提供されれば、メインアプリケーション153は再構築する必要がなく、より性能が高いエンコードカードをそのまま利用（制御）することができる。

【0163】

メインアプリケーション153と音楽データ入出力カプセル155との間、並びに、エンコード制御アプリケーション152と制御データ入出力カプセル154との間においても、同様のことが言える。

【0164】

なお、以上においては、エンコーダ112が、ATRAC1対応エンコード処理およびMPEG1対応エンコード処理を実行する場合を例として説明したが、MP3、その他のフォーマットのエンコード処理やエフェクト機能などを実行させるようにすることもできる。

【0165】

例えば、エンコーダ 112 は、ATRAC3 (Adaptive Transform Acoustic Coding 3) (商標), MPEG-2AAC (Advanced Audio Coding) (商標), ODesign Music Codec (商標), TwinVO (Transform-Domain Weighted Interleave Vector Quantization) (商標), MS Audio (Microsoft Audio(WMA:Windows Media Audio)) (商標), Ogg Vorbis (商標) などのフォーマットのエンコーダとすることも可能である。

【0166】

次に、図 16 を参照して、ステートの異常時の初期化処理について説明する。図 16 におけるステート Q は、図 9 におけるレディステート (READY) E、レシーブステート (RECEIVE) F, L、リプライステート (REPLY) G, M、マネージステート (MANAGE) H, K、センドステート (SEND) I, J に対応している。すなわち、これらのいずれのステートからもインタラプト (INTERRUPT) のメッセージが受信された場合の初期化のルートとして、イニシャライズ 2 ステート (INITIALIZE 2) D、およびレディステート (READY) E の第 1 のルート、ターミネート 1 ステート (TERMINATE 1) N、イニシャライズ 1 ステート (INITIALIZE 1) C、イニシャライズ 2 ステート (INITIALIZE 2) D、およびレディステート (READY) E の第 2 のルート、またはターミネート 1 ステート (TERMINATE 1) N、ターミネート 0 ステート (TERMINATE 0) O、およびエグジットステート (EXIT) P の第 3 のルート、の 3 つのルートが用意されている。

【0167】

図 17 は、この異常時における初期化処理を説明するフローチャートである。この処理は、インタラプトのメッセージを受信したとき、開始される。ステップ S351 において、各プロセスは、インタラプトのメッセージとして (INT, QUT) が受信されたか否かを判定する。受信したインタラプトのメッセージが (INT, QUT) であると判定された場合、ステップ S352 に進み、プロセスは、イニシャライズ 2 ステート (INITIALIZE 2) D に遷移する。このステートにおいては、図 10 のステップ S204 の処理における場合と同様に、第 1 のバッファの初期化処理が行われる。その後、レディステート (READY) E に遷移する。

【0168】

このように、プロセスを終了せず（エグジットステート（EXIT）Pに遷移せず）、第1のバッファの初期化のみを行い、レディステート（READY）Eに遷移するようにすることより、例えば、一旦終了させてしまったプロセスを、エンコード処理のために、再度、生成することによるオーバーヘッドの発生を、防止することができる。

【0169】

また、何らかの異常が起きた場合に、第1のバッファの初期化を行うだけで、その異常が回復することが期待される。この第1のパスの処理は、後述する第2のパスより短い時間で完了することができるので、より迅速に、異常な事態を回復させることが可能となる。

【0170】

ステップS351において、インタラプトを受けたメッセージが（INT, QUT）メッセージではないと判定された場合、ステップS353に進み、プロセスは、インタラプトを受けたメッセージが（INT, RST）メッセージであるか否かを判定する。インタラプトを受けたメッセージが（INT, RST）メッセージである場合には、ステップS354に進み、プロセスは、ターミネート1ステート（TERMINATE1）Nに遷移する。このステートでは、第2のバッファの開放処理が行われる。

【0171】

次に、ステップS355に進み、プロセスは、イニシャライズ1ステート（INITIALIZE1）Cに遷移する。このステートにおいては、図10のステップS203の処理における場合と同様に、第2のバッファの確保、第2のバッファの初期化、並びにレジスタの初期化処理が実行される。

【0172】

次に、ステップS356に進み、プロセスは、イニシャライズ2ステート（INITIALIZE2）Dに遷移する。ここでは、ステップS352における場合と同様に、第1のバッファの初期化処理が実行される。

【0173】

その後、プロセスは、レディステート（READY）Eに遷移する。

【0174】

このように、第2のパスの初期化処理においては、第1のパスにおける初期化処理のステートを含む、それ以外のステートの初期化処理が行われるため、第1のパスの初期化処理を行う場合に比べて、より長い時間がかかるが、より確実に、異常な事態を回復することが可能となる。

【0175】

ステップS353において、インタラプトを受けたメッセージが(INT, RST)メッセージではないと判定された場合、そのメッセージは、結局(INT, TRM)メッセージであるということになる。そこで、この場合、ステップS357に進み、プロセスは、ターミネート1ステート(TERMINATE1)Nに遷移する。そこで、ステップS354における場合と同様に、第2のバッファの開放処理が行われる。次に、ステップS358に進み、プロセスは、ターミネート0ステート(TERMINATE0)Oに遷移する。このステートにおいては、第1のバッファの開放処理が行われるとともに、デバイスをクローズする処理が行われる。

【0176】

そして、エグジットステート(EXIT)Pにプロセスは遷移し、リアルタイムOSが終了され、電源オフの状態になる。

【0177】

この第3のパスは、第1および第2のパスの初期化処理では回復することができなかった場合に行われる。この場合、リアルタイムOSが再起動され、各プロセスが新たに再び生成される処理が行われることになるので、第1のパスまたは第2のパスの初期化処理に比べ、より長い時間がかかるが、より確実に、異常事態を回復することが可能となる。

【0178】

上述した異常時における3つのパスの初期化処理を時系列に沿って示すと、図18に示すようになる。

【0179】

例えば、メインアプリケーション153は、異常が発生した場合、ステップS411において(INT, QUT)メッセージをエンコード処理マネージャ181に出

力する。エンコード処理マネージャ181は、この (INT, QUT) メッセージを受信すると、ステップS421においてイニシャライズ2ステート (INITIALIZE2) Dに遷移し、第1のバッファの初期化処理を実行する。ステップS422において、エンコード処理マネージャ181は、レディステート (READY) Eに遷移する。エンコード処理マネージャ181は、この第1のパスの初期化処理により、異常が回復したか否かをステップS423において、メインアプリケーション153に通知する。

【0180】

メインアプリケーション153は、第1のパスの初期化処理により、異常が回復していない場合には、さらに、ステップS412において、(INT, RST) メッセージをエンコード処理マネージャ181に出力する。エンコード処理マネージャ181は、メインアプリケーション153から (INT, RST) メッセージを受信すると、ステップS424において、ターミネート1ステート (TERMINATE1) Nに遷移し、第2のバッファの開放処理を行う。次に、ステップS425において、エンコード処理マネージャ181は、イニシャライズ1ステート (INITIALIZE1) Cに遷移し、第2のバッファの確保、第2のバッファの初期化、並びにレジスタの初期化処理を実行する。

【0181】

ステップS426において、エンコード処理マネージャ181は、イニシャライズ2ステート (INITIALIZE2) Dに遷移し、第1のバッファの初期化処理を実行する。ステップS427において、エンコード処理マネージャ181は、レディステート (READY) Eに戻る。エンコード処理マネージャ181は、ステップS428において、以上の第2のパスの初期化処理により、異常が回復したか否かをメインアプリケーション153に通知する。

【0182】

メインアプリケーション153は、以上の第2のパスの初期化処理により、エンコード処理マネージャ181の異常がまだ回復しないと判定した場合、ステップS413において、エンコード処理マネージャ181に対して (INT, TRM) メッセージを出力する。

【0183】

エンコード処理マネージャ181は、メインアプリケーション153から（IN T, TRM）メッセージを受信すると、ステップS429において、ターミネート1ステート（TERMINATE1）Nに遷移し、第2のバッファの開放処理を行う。ステップS430において、エンコード処理マネージャ181は、さらに、ターミネート0ステート（TERMINATE0）Oに遷移し、第1のバッファの開放処理とデバ イスクローズ処理を実行する。

【0184】

ステップS431において、エンコード処理マネージャ181は、レディステート（READY）Eに遷移し、以上の第3のパスの初期化処理により、異常が回復したか否かを、ステップS432において、メインアプリケーション153に通知する。

【0185】

メインアプリケーション153は、以上の第3のパスの初期化処理により、まだ異常が回復していないと判定した場合、ステップS414において、リアルタイムOSに再起動を要求する。

【0186】

リアルタイムOSは、ステップS401において、メインアプリケーション153からの要求に基づいて、再起動処理を実行する。

【0187】

そして、ステップS402において、リアルタイムOSは、再起動後、再びエンコード処理マネージャ181を生成する処理を実行する。これにより、上述した図10のステップ201乃至S205の処理により、イグゼキュートステート（EXECUTE）A、イニシャライズ0ステート（INITIALIZE0）B、イニシャライズ1ステート（INITIALIZE1）C、イニシャライズ2ステート（INITIALIZE2）Dの各ステートを経て、レディステート（READY）Eに遷移する処理が実行される。

【0188】

このように、3つのパスの初期化処理を設け、より短い処理時間ですむパスか

ら初期化処理を順次行うようにしたので、より迅速に、異常を回復することが可能となる。

【0189】

レシーブステート (RECEIVE) F, L、センドステート (SEND) I, J 以外のステートにおいては、基本的にメッセージが受信できない状態となっている。従って、そのようなステートにおいて、例えば、異常が発生し、そのステートにおける処理が完了できなくなってしまったような場合、異常な事態を回復することができなくなってしまうおそれがある。また、レシーブステート (RECEIVE) F, L、並びにセンドステート (SEND) I, J においても異常が発生し、所定のメッセージを受信したとしても、他のステートに遷移できなくなってしまうような場合がある。このような異常な状態が発生した場合にも、確実に、異常を回復することができるようにするために、各ステートには、上述したように、インタラプトのメッセージを受け付けることができるようになされている。

【0190】

以上のように、プロセス間通信用メッセージのヘッダに、MSGT=INT (INTERRUPT) を設定することにより、UNIX の SIGNAL システムコールのように、例外信号 (非同期信号) により、プロセスのステートをイニシャライズステート (INITIALIZE) (イニシャライズステート (INITIALIZE) C, D) やターミネートステート (TERMINATE) (ターミネートステート (TERMINATE) N, O) に遷移させることができる。その結果、環境変数などを予め退避させておくことより、遷移前のステートへ確実に戻ることが可能となる。

【0191】

図 19 は、本発明を適用した音楽配信サービスシステムの配信側の他の構成例を示している。なお、図中、図 2 における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【0192】

端末 200 は、アラーム LAN 101 およびメディアム LAN 102 に接続されており、サーバ 113 から、PCM 非圧縮音楽データを受け取り、それをエンコードして、サーバ 113 に供給する。すなわち、端末 200 は、図 2 に示す端末 111

の機能に加え、エンコーダ 1 1 2 の機能をさらに備えている。

【 0 1 9 3 】

図 2 0 は、端末 2 0 0 の構成例を表している。この端末 2 0 0 は、例えばコンピュータで構成される。CPU 5 1 1 にはバス 5 1 5 を介して入出力インターフェース 5 1 6 が接続されており、CPU 5 1 1 は、入出力インターフェース 5 1 6 を介して、ユーザから、キーボード、マウスなどよりなる入力部 5 1 8 から指令が入力されると、例えば、ROM 5 1 2、ハードディスク 5 1 4、またはドライブ 5 2 0 に装着される磁気ディスク 5 3 1、光ディスク 5 3 2、光磁気ディスク 5 3 3、若しくは半導体メモリ 5 3 4 などの記録媒体に格納されているプログラムを、RAM 5 1 3 にロードして実行する。さらに、CPU 5 1 1 は、その処理結果を、例えば、入出力インターフェース 5 1 6 を介して、LCD などよりなる表示部 5 1 7 に必要に応じて出力する。

【 0 1 9 4 】

なお、プログラムは、ハードディスク 5 1 4 や ROM 5 1 2 に予め記憶しておき、端末 2 0 0 と一体的にユーザに提供したり、磁気ディスク 5 3 1、光ディスク 5 3 2、光磁気ディスク 5 3 3、半導体メモリ 5 3 4 等のパッケージメディアとして提供したり、衛星、ネットワーク等から通信部 5 1 9 を介してハードディスク 5 1 4 に提供することができる。

【 0 1 9 5 】

半導体メモリ 5 3 4 は、フラッシュメモリなどの不揮発性メモリであることが望ましい。また、半導体メモリ 5 3 4 を内蔵するパッケージメディアは、マイクロコンピュータを内蔵し半導体メモリ 5 3 4 に対する読み書きの認証が可能なものであることが望ましい。例えば、メモリスティック（商標）、SDメモリーカード（商標）、コンパクトフラッシュ（商標）、スマートメディア（商標）、マルチメディアカード（商標）、マイクロドライブ（商標）、IDフォーマット（商標）、Thumb Drive（商標）などとすることができる。

【 0 1 9 6 】

図 2 1 は、端末 2 0 0 の RAM 5 1 3 にロードされ、CPU 5 1 1 により実行されるプログラムの構成例を示している。この端末 2 0 0 には、図 2 の端末 1 1 1 の制

御プロセスWの他、図6のエンコーダ112の音楽データ入出力カプセル155およびエンコード処理カプセル156がさらに設けられている。端末200が、このような構成を有することより、端末111と同様のエンコード処理の実行が可能となる。

【0197】

この場合においても、制御プロセスWに対してデータ入出力マネージャ171が行う処理は、音楽データ入出力カプセル155としてカプセル化されており、同様に、エンコード処理マネージャ181が実行する処理は、エンコード処理カプセル156としてカプセル化されている。

【0198】

また、この場合においても、上述した状態遷移図に対応して、異常時における初期化処理を含む各種の処理が実行される。その処理は、上述した場合と基本的に同様であるので、その説明は省略する。

【0199】

なお、本明細書において、媒体により提供されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理はもちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

【0200】

また、本明細書において、システムの用語は、複数の装置、手段などより構成される全体的な装置を意味するものとする。

【0201】

【発明の効果】

本発明によれば、ハードウェアが変更されたとしても、制御部を変更する必要がなくなるか、変更するとしても、わずかの変更で済み、1つの制御部で、多くのハードウェアを共通に使用することが可能になる。

【図面の簡単な説明】

【図1】

従来のエンコーダの機能的構成例を示すブロック図である。

【図 2】

本発明を適用した音楽配信サービスシステムの構成例を示す図である。

【図 3】

図 2 のエンコーダの構成例を示す図である。

【図 4】

サバイバルタイムを説明する図である。

【図 5】

リアルタイム OS の原理を説明する図である。

【図 6】

図 2 のエンコーダの機能的構成例を示す図である。

【図 7】

図 6 におけるプロセスの ATRAC エンコード処理を説明する図である。

【図 8】

図 6 におけるプロセスの MPEG エンコード処理を説明する図である。

【図 9】

図 6 のプロセスのステートの遷移を説明する図である。

【図 10】

正常時における初期化処理を説明するフローチャートである。

【図 11】

プロセス間通信用メッセージの構成を説明する図である。

【図 12】

プロセス間通信用メッセージの他の構成を説明する図である。

【図 13】

プロセスの処理を説明するフローチャートである。

【図 14】

図 6 のメインアプリケーションの動作を説明するフローチャートである。

【図 15】

図 6 のエンコード処理マネージャの動作を説明するフローチャートである。

【図 16】

異常時の初期化処理時におけるプロセスのステートの遷移を説明する図である。

【図 17】

異常時の初期化処理を説明するフローチャートである。

【図 18】

異常時の初期化処理を説明するアローチャートである。

【図 19】

本発明を適用した音楽配信サービスシステムの他の構成例を示す図である。

【図 20】

図 19 の端末の構成例を示すブロック図である。

【図 21】

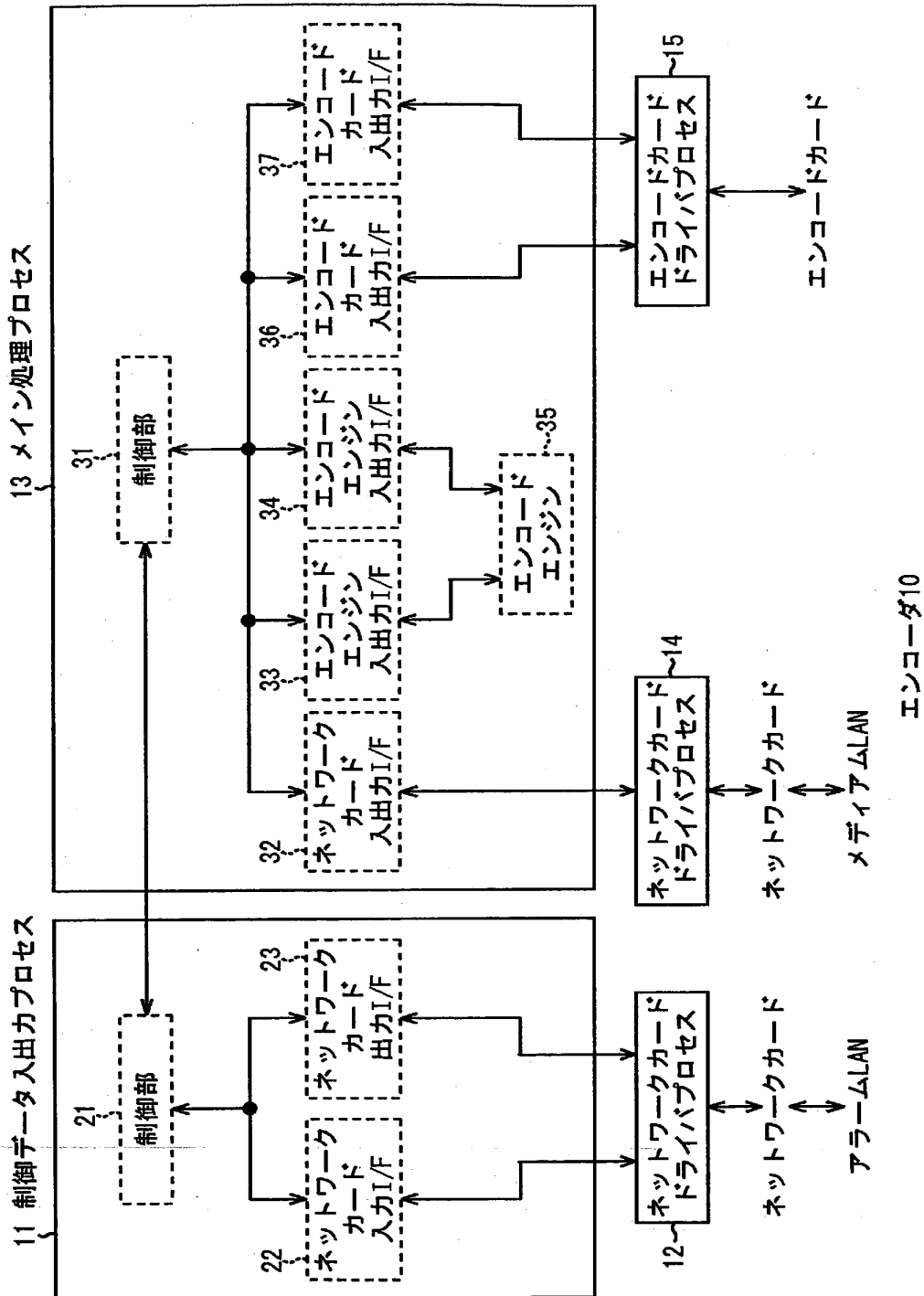
図 19 の端末の機能的構成例を示すブロック図である。

【符号の説明】

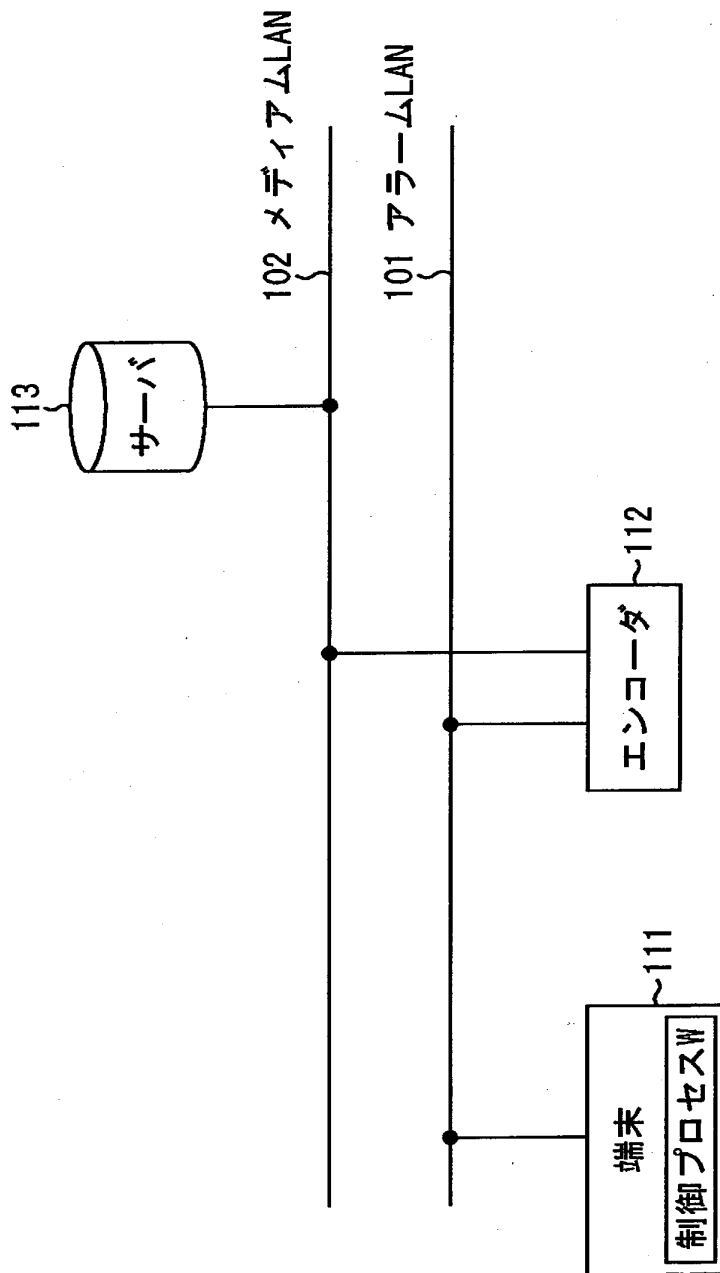
101 アラームLAN, 102 メディアムLAN, 111 端末, 112
エンコーダ, 113 サーバ, 121 CPU, 122 ROM, 123
RAM, 124, 125 ネットワークカード, 126 エンコードカード,
127 ハードディスク, 151 カプセルマネージャ, 152 エンコ
ード制御アプリケーション, 153 メインアプリケーション, 154 制
御データ入出力カプセル, 155 音楽データ入出力カプセル, 156 エ
ンコード処理カプセル, 161 エンコード制御マネージャ, 162 ネッ
トワークカード入力I/F, 163 ネットワークカード出力I/F, 164 ネ
ットワークカードドライバ, 171 データ入出力マネージャ, 172 ネ
ットワークカード入出力I/F, 173 ネットワークカードドライバ, 18
1 エンコード処理マネージャ, 182, 183 エンコードエンジン入出力
I/F, 184 エンコードエンジン, 185 エンコードカード入出力I/F,
186 エンコードカード入出力I/F, 187 エンコードカードドライバ

【書類名】 図面

【図 1】

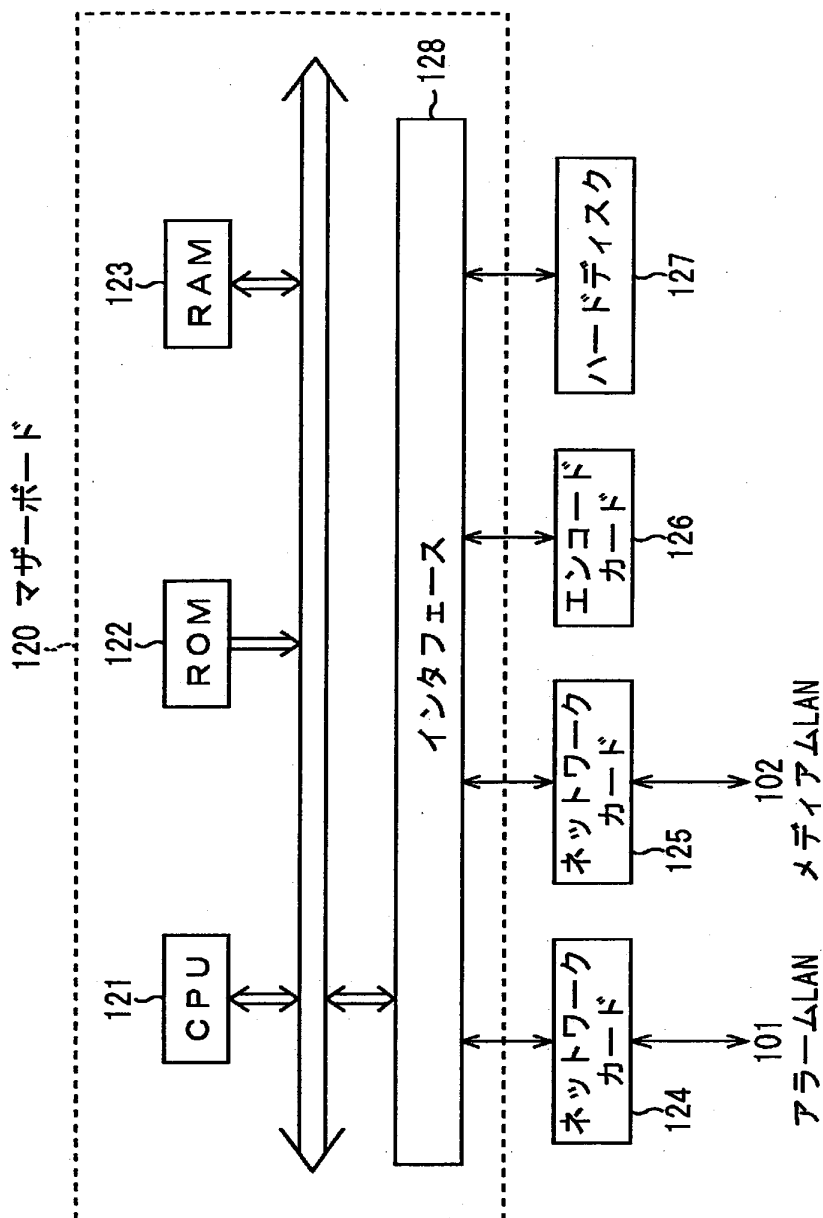


【図2】



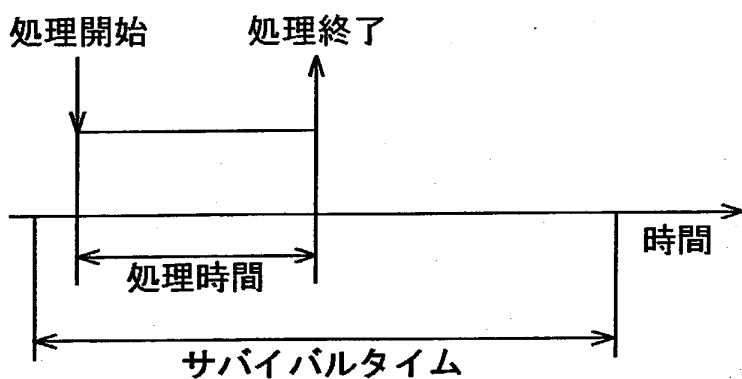
音楽配信サービスシステム

【図3】

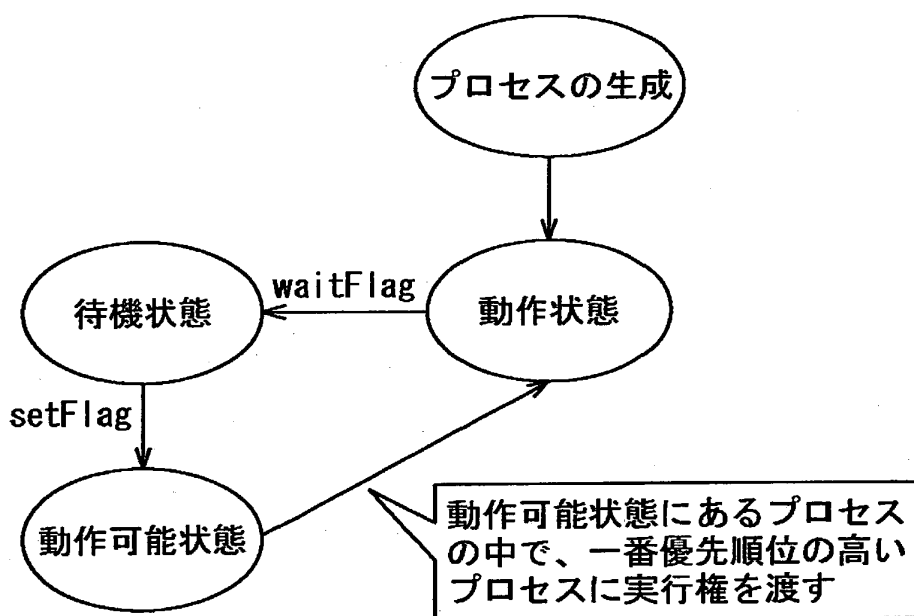


エンコーダ 112

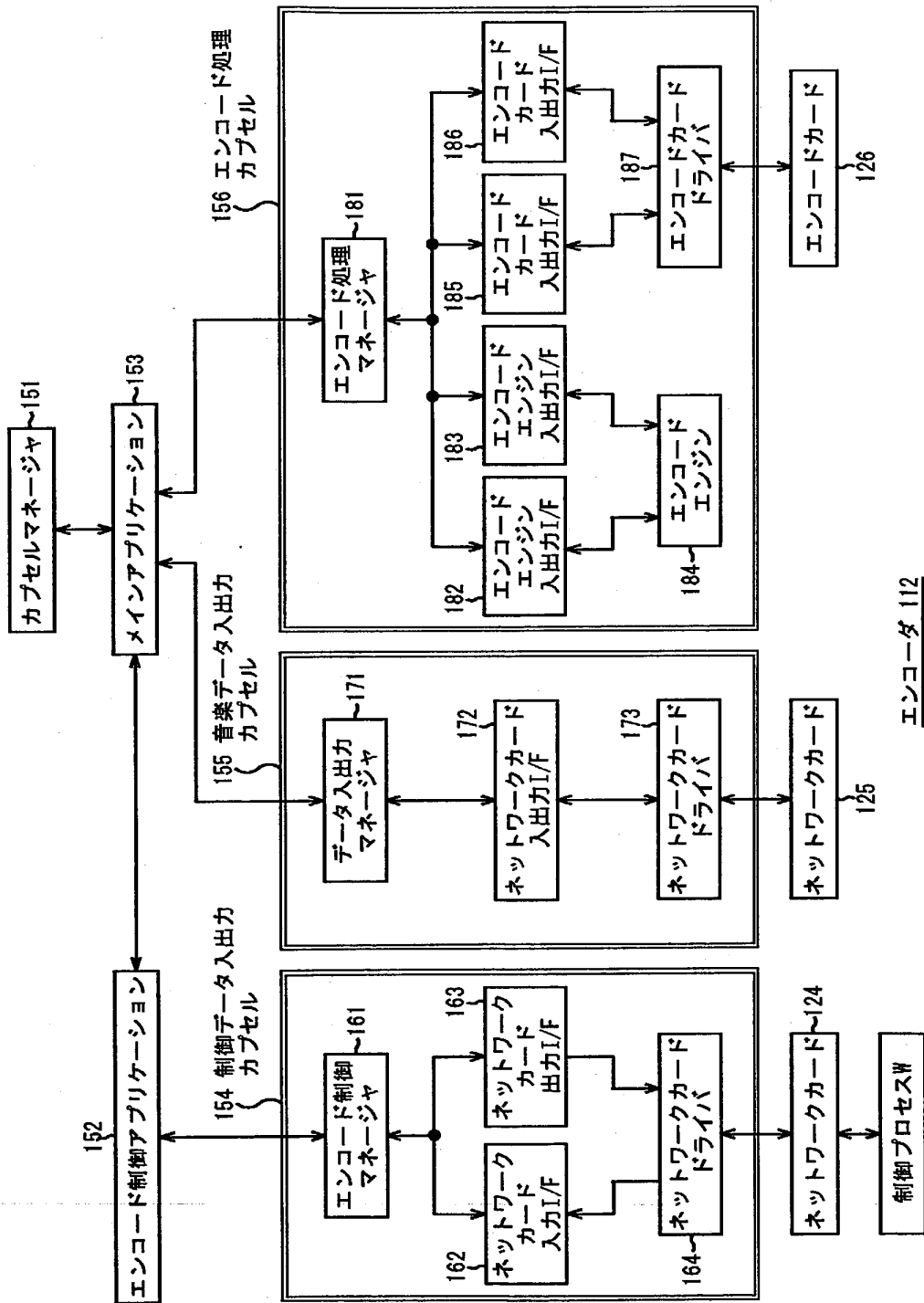
【図 4】



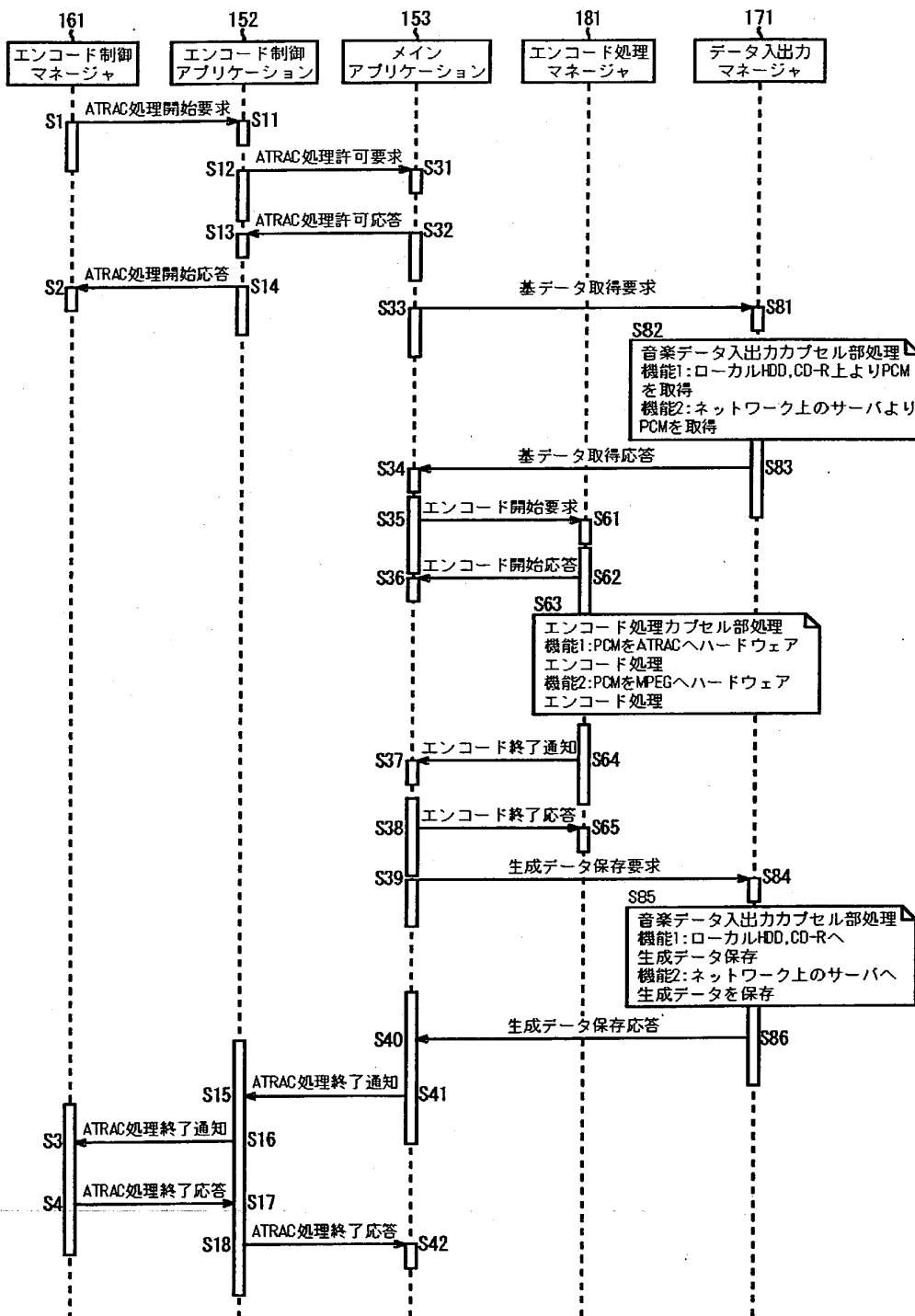
【図 5】



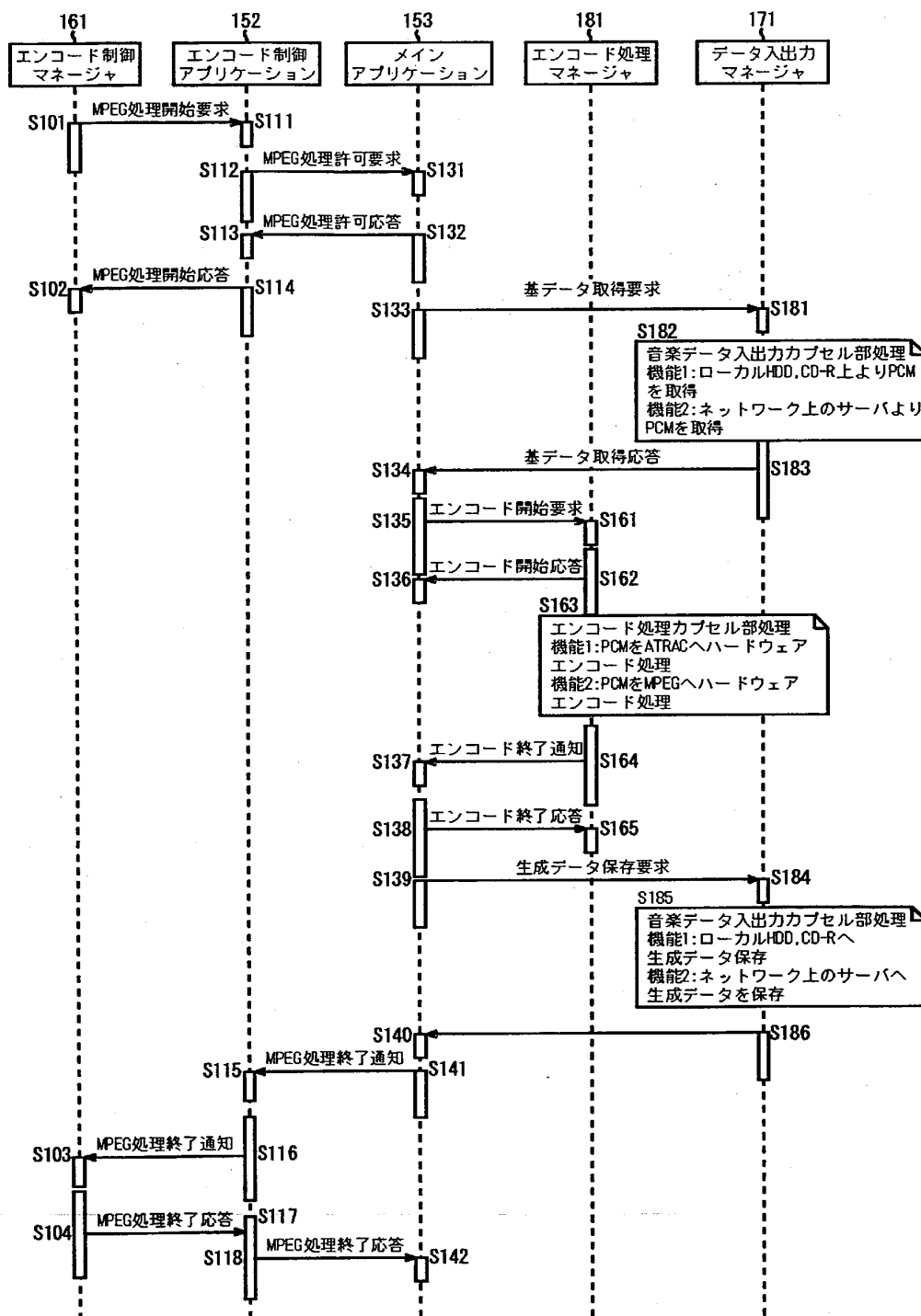
【図6】



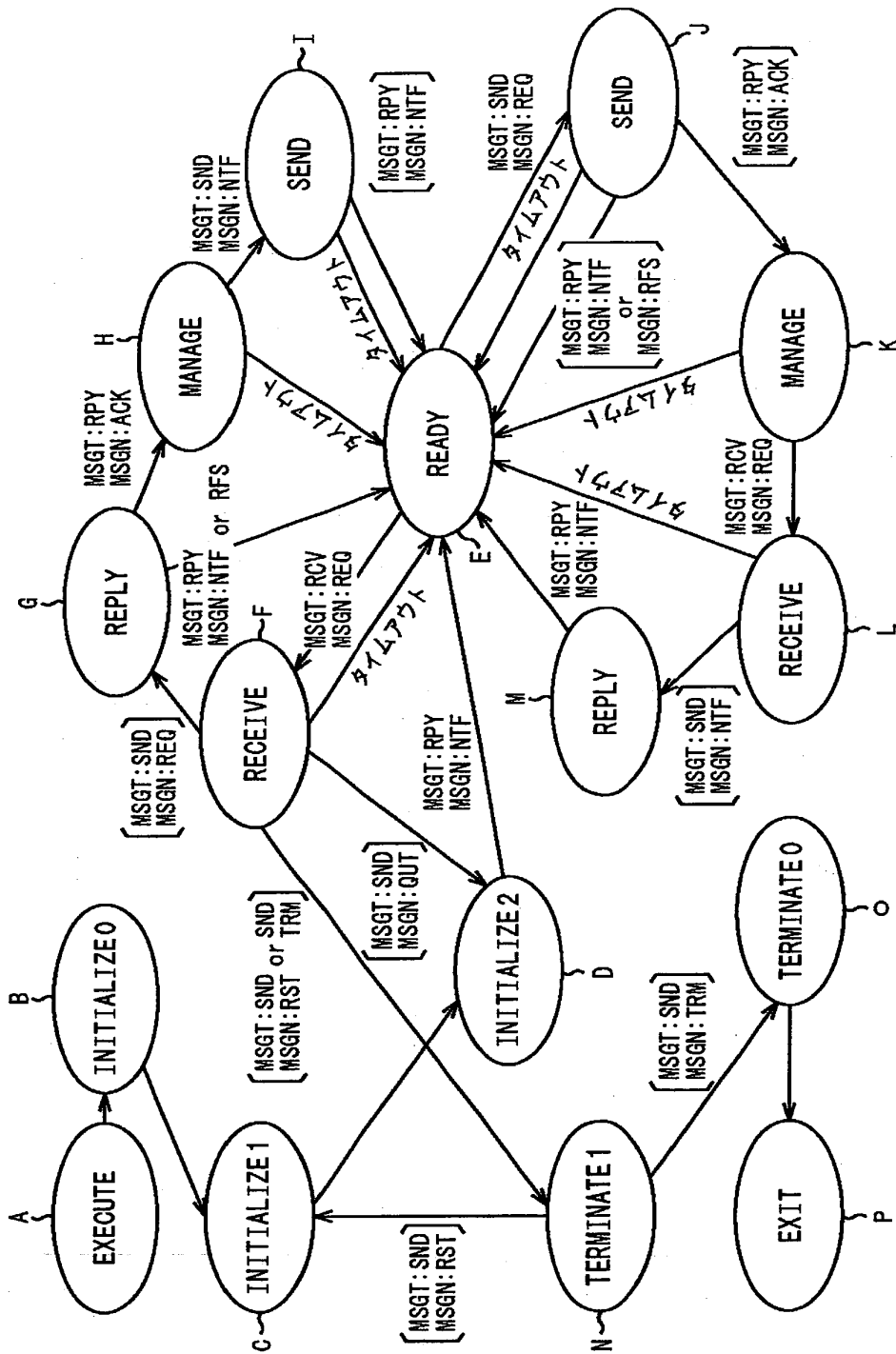
【図 7】



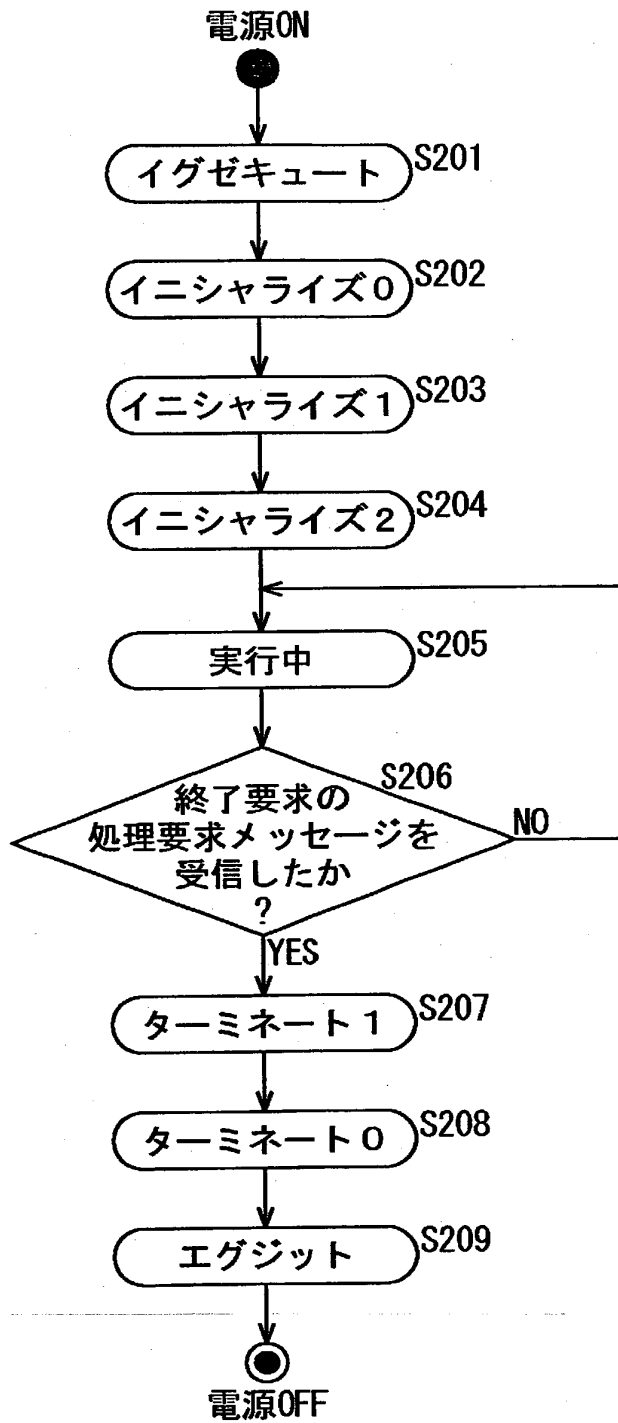
【図 8】



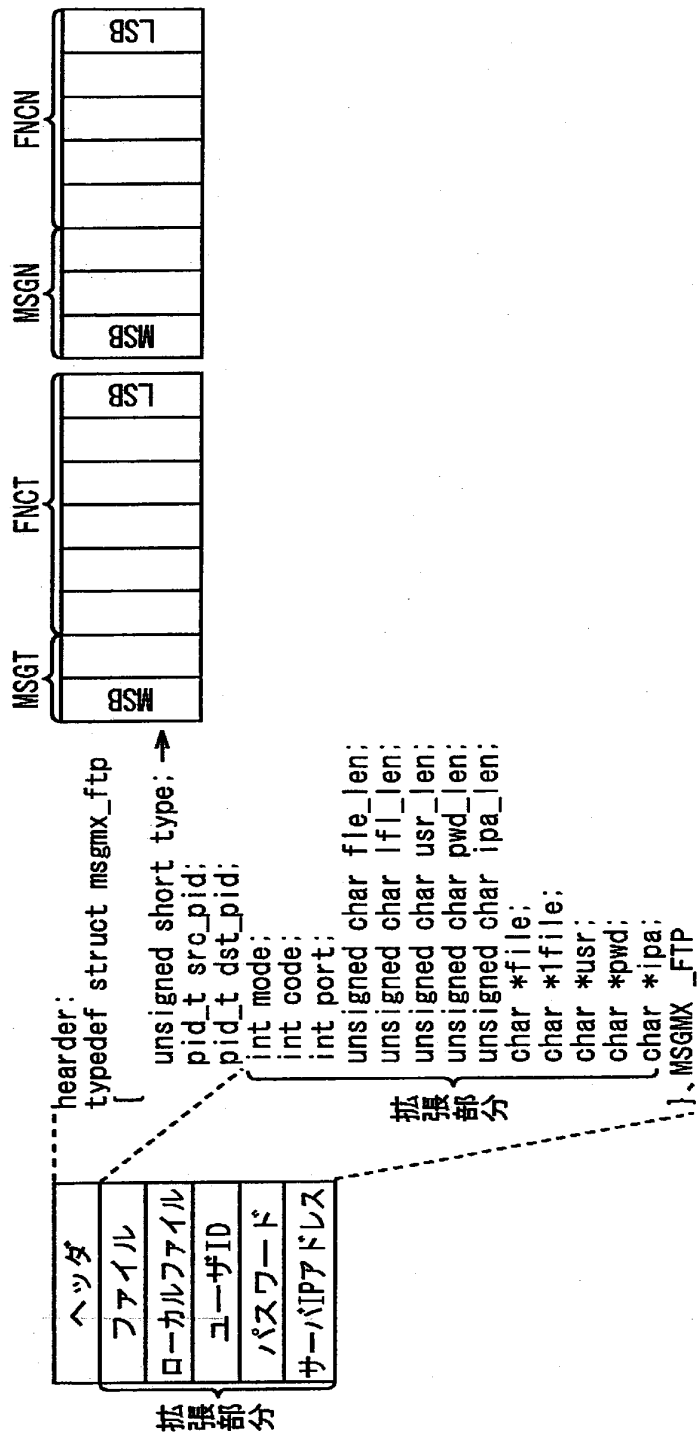
【図9】



【図10】



【図11】

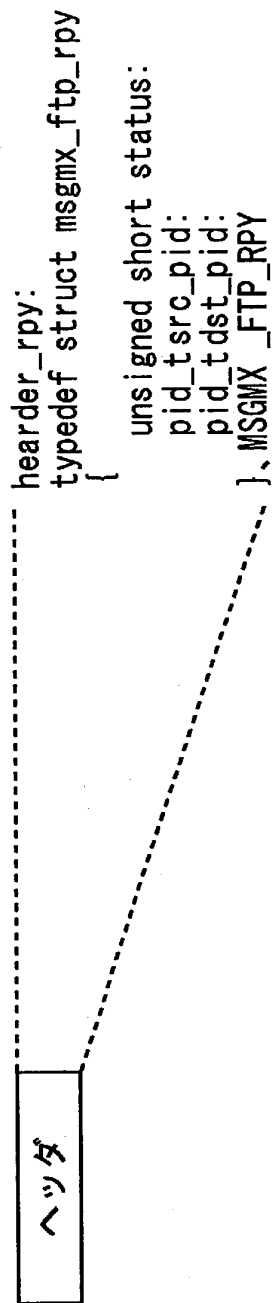


(A)

(B)

(C)

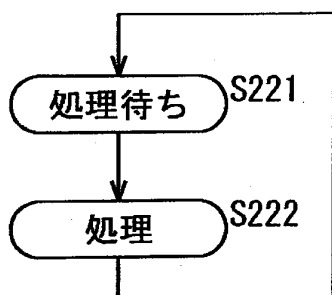
【図12】



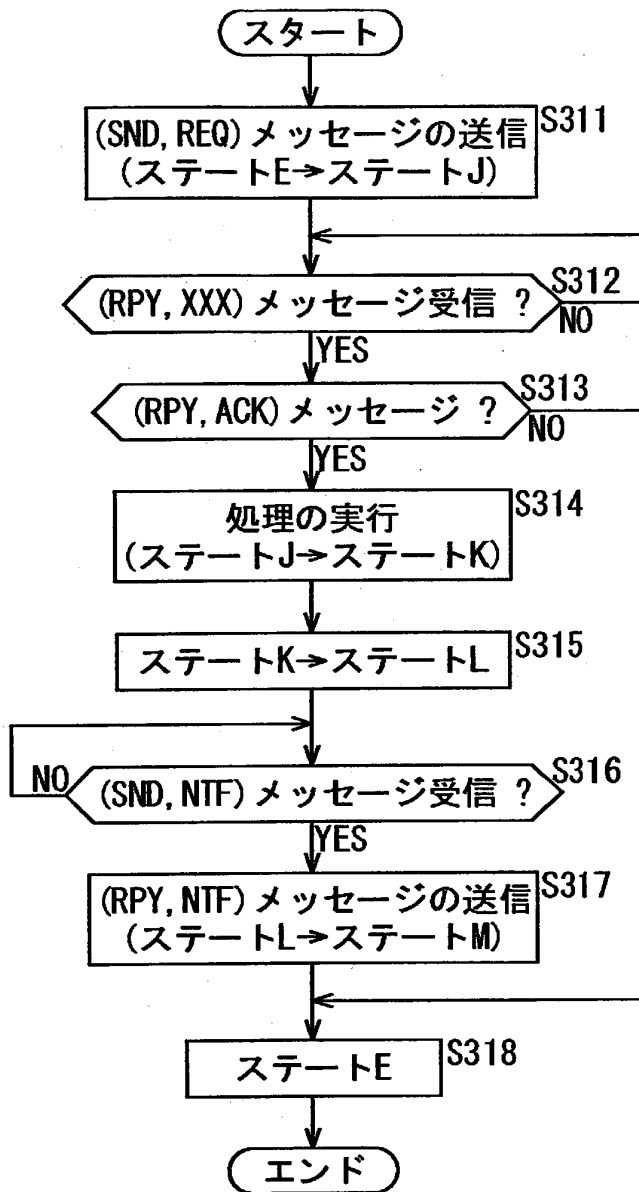
(B)

(A)

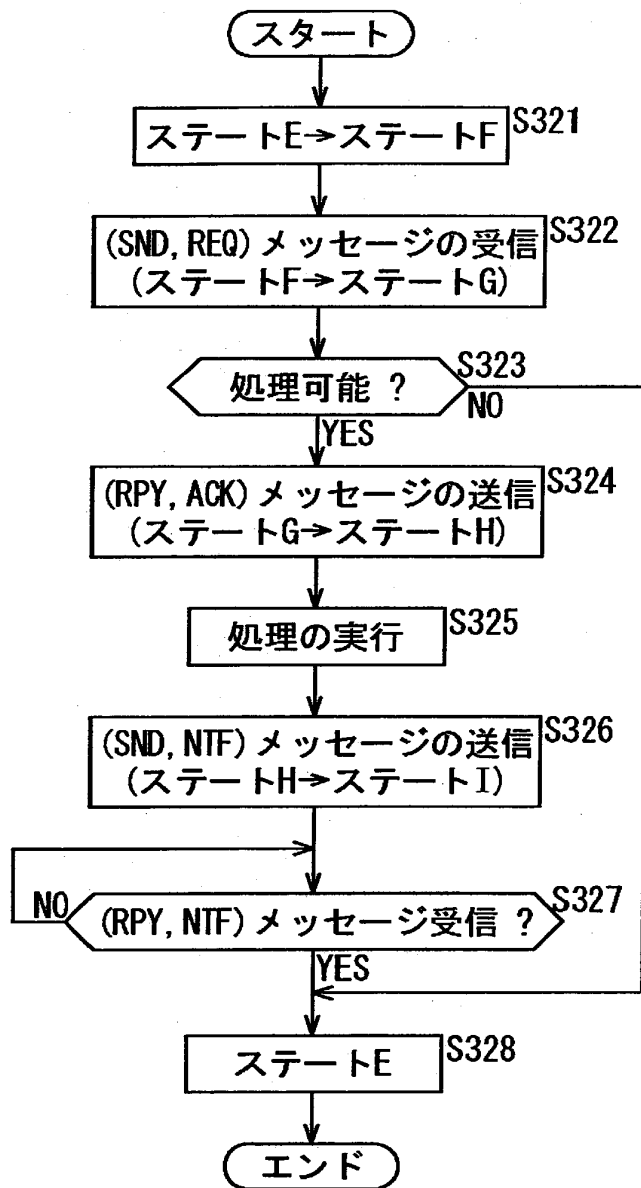
【図 1 3】



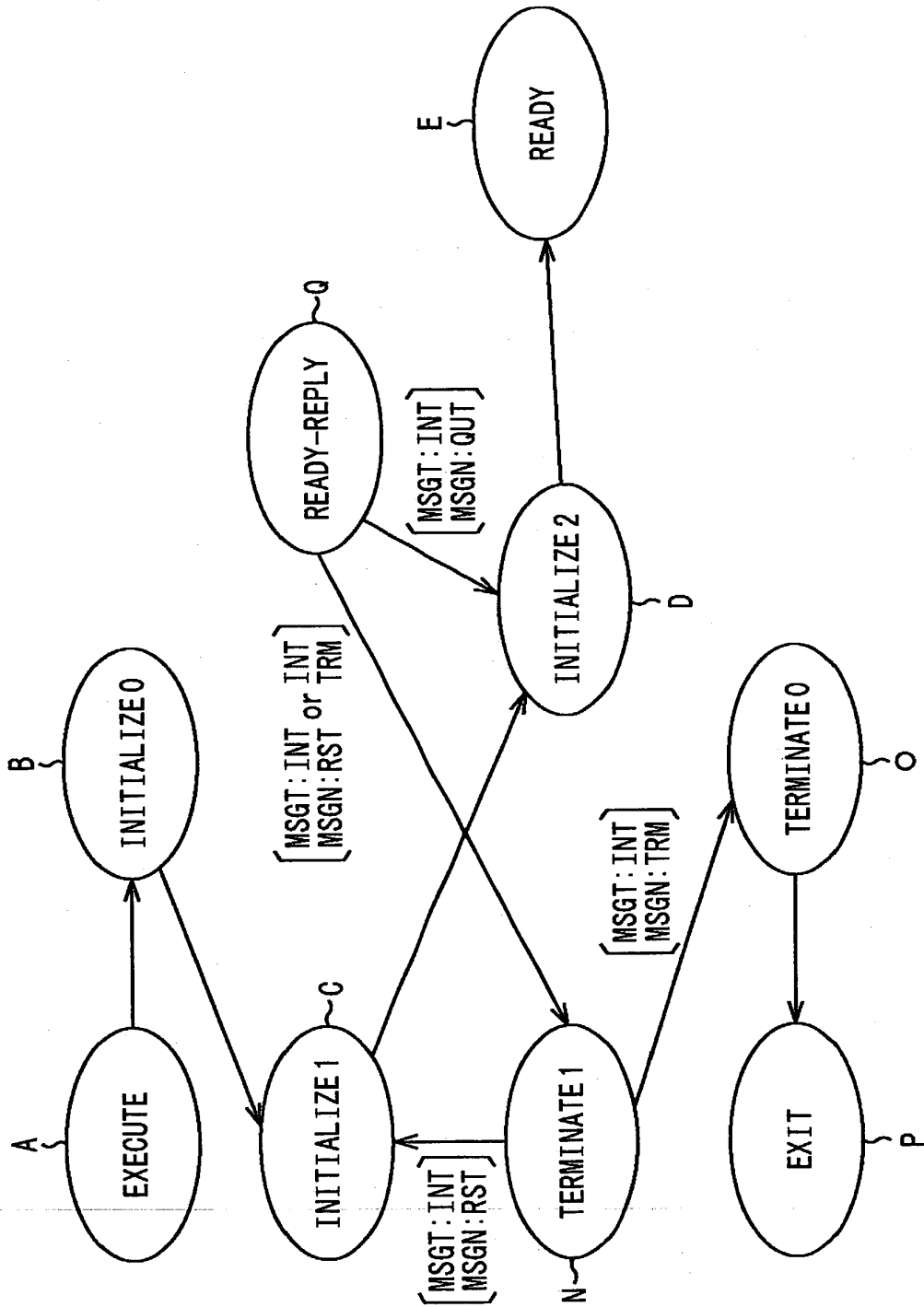
【図 1 4】



【図15】

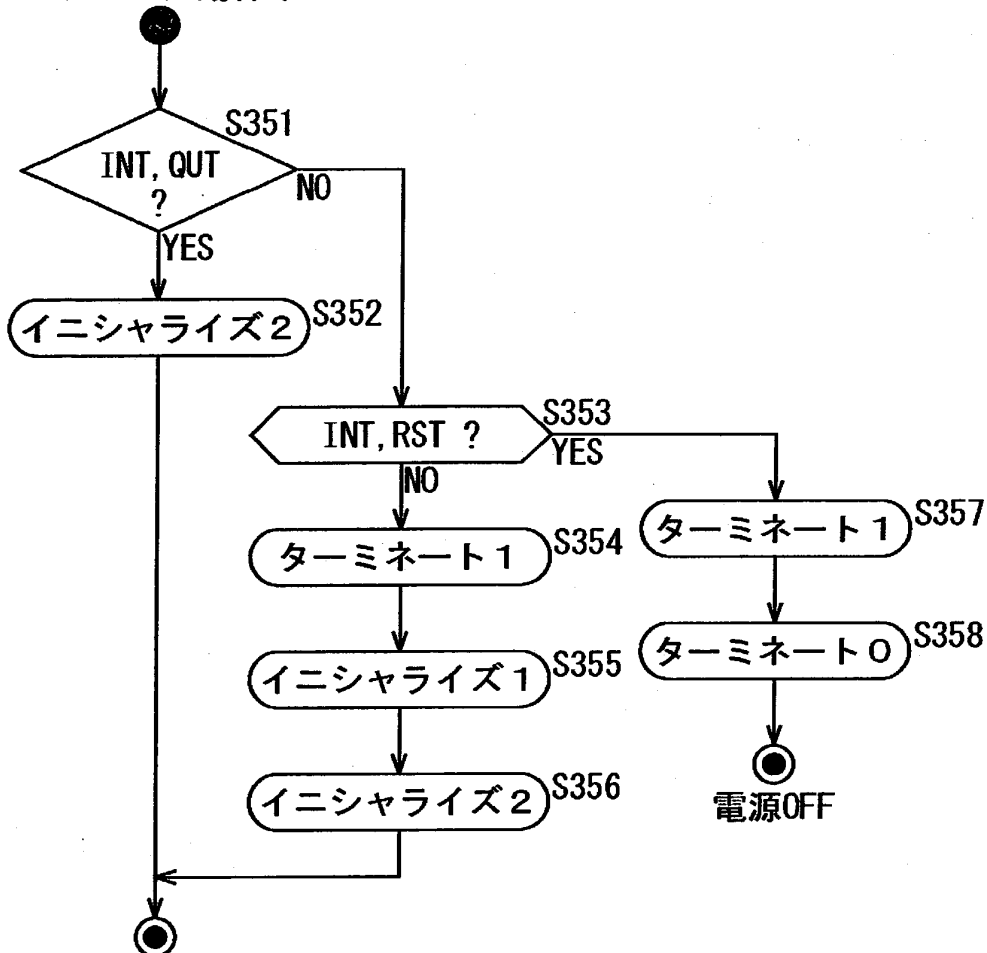


【図16】



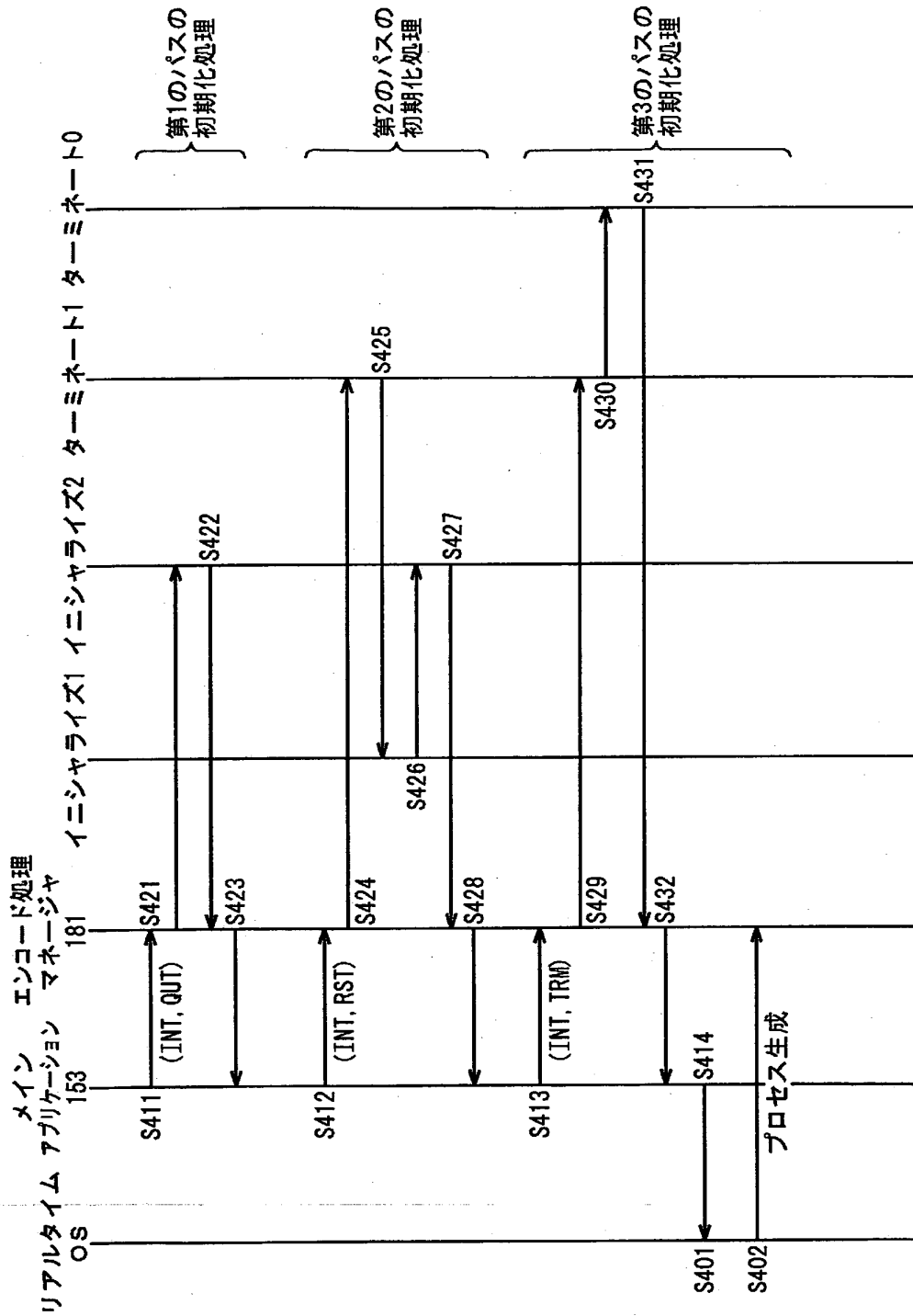
【図17】

モジュール動作中

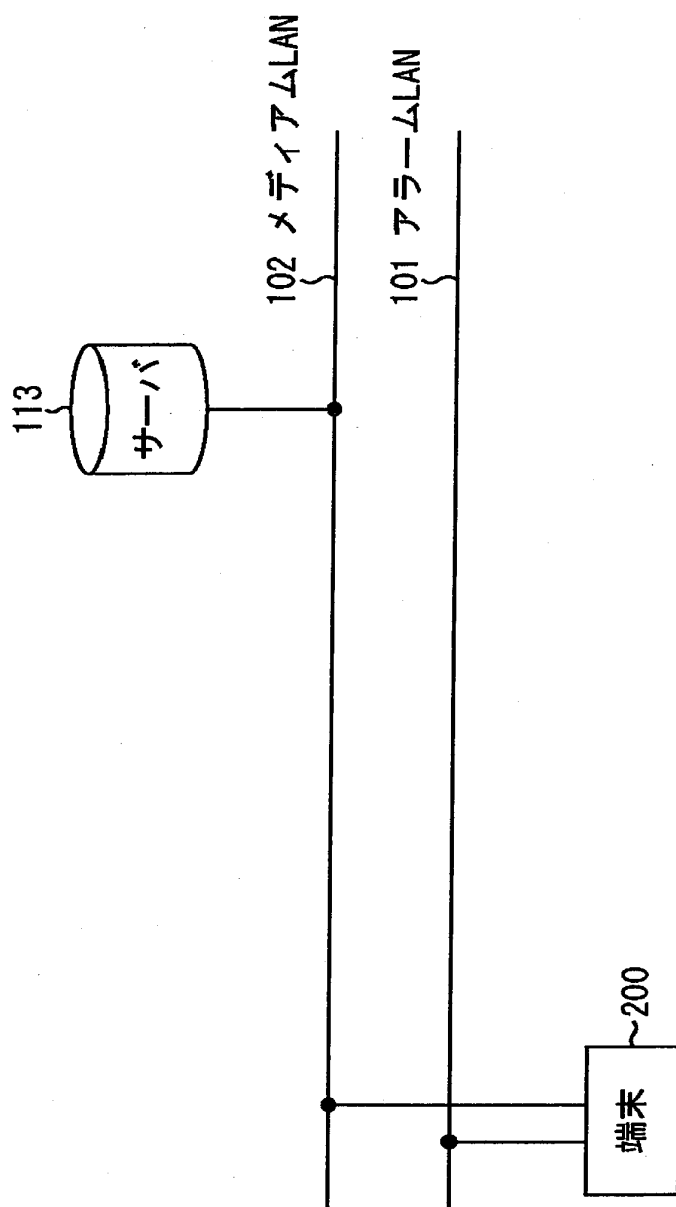


モジュール準備完了

【図18】

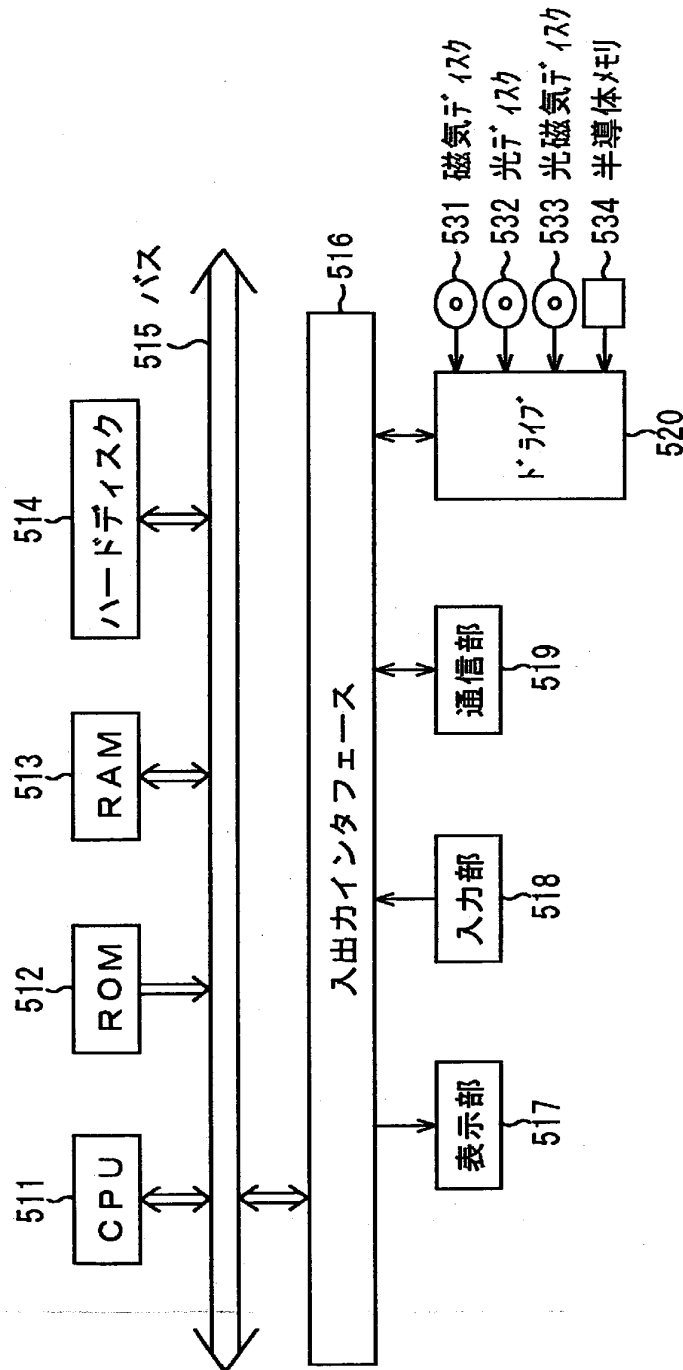


【図19】



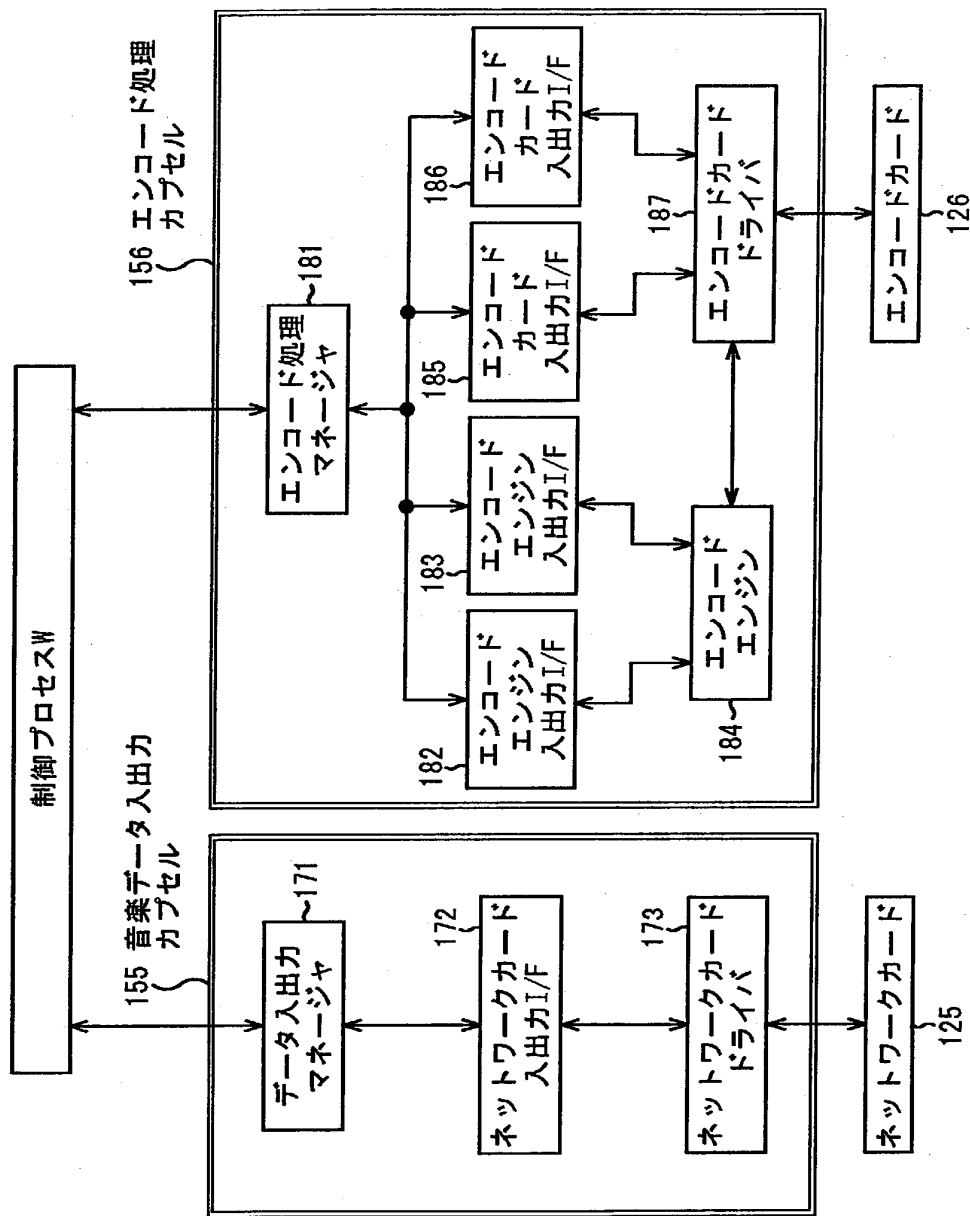
音楽配信サービスシステム

【図20】



端末 200

【図 21】



端末 200

【書類名】 要約書

【要約】

【課題】 エンコードのためのハードウェアの変更に容易に対応することができるようにする。

【解決手段】 エンコード処理を実行するハードウェアとしてのエンコードカード126を制御するソフトウェアとしてのエンコード処理マネージャ181乃至エンコードドライバ187を、エンコード処理カプセル156としてカプセル化することより、メインアプリケーション153は、例えば、エンコードする音楽データが格納されている場所、エンコードされた音楽データが格納される場所、またはエンコード処理の種類などをエンコード処理マネージャ181に通知するだけで、所定のエンコード処理を実行させることができる。

【選択図】 図6

特 2000-325218

認定・付加情報

特許出願の番号	特願 2000-325218
受付番号	50001378024
書類名	特許願
担当官	第七担当上席 0096
作成日	平成 12 年 10 月 30 日

<認定情報・付加情報>

【特許出願人】

【識別番号】	000002185
【住所又は居所】	東京都品川区北品川 6 丁目 7 番 35 号
【氏名又は名称】	ソニー株式会社

【代理人】

申請人	
【識別番号】	100082131
【住所又は居所】	東京都新宿区西新宿 7 丁目 5 番 8 号 GOWA 西 新宿ビル 6 F 稲本国際特許事務所
【氏名又は名称】	稲本 義雄

次頁無

特2000-325218

出願人履歴情報

識別番号

[000002185]

1. 変更年月日

1990年 8月30日

[変更理由]

新規登録

住 所

東京都品川区北品川6丁目7番35号

氏 名

ソニー株式会社

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

THIS PAGE BLANK (USPTO)